US005870146A

United States Patent [19]

Zhu

[11] Patent Number: 5,870,146

[45] Date of Patent: Feb. 9, 1999

[54] **DEVICE AND METHOD FOR DIGITAL VIDEO TRANSCODING**

[75] Inventor: **Qin-Fan Zhu**, Wilmington, Mass.

[73] Assignee: **MultiLink, Incorporated**, Andover, Mass.

[21] Appl. No.: **786,070**

[22] Filed: **Jan. 21, 1997**

[51] **Int. Cl.**[6] **H04N 7/24**; H04N 7/32

[52] **U.S. Cl.** **178/405**; 348/409; 375/240

[58] **Field of Search** 348/27, 384, 390, 348/400–403, 405, 409–413, 415, 420, 699; 382/232, 236, 238, 248, 250–251; 375/240, 245; H04N 7/130, 7/137

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,411,001	10/1983	Van Buul et al.	375/30
4,972,260	11/1990	Fujikawa et al.	358/136
5,072,295	12/1991	Murakami et al.	358/136
5,202,685	4/1993	Nagata	341/76
5,204,677	4/1993	Akagiri et al.	341/118
5,283,814	2/1994	Iwahashi et al.	375/27
5,333,012	7/1994	Singhal et al.	348/405
5,349,383	9/1994	Parke et al.	348/397
5,355,378	10/1994	Ohta	371/53
5,412,431	5/1995	Vogel	348/405
5,517,142	5/1996	Hamilton	348/405
5,537,440	7/1996	Eyuboglu et al.	375/245
5,541,852	7/1996	Eyuboglu et al.	395/200.62
5,544,266	8/1996	Koppelmans et al.	382/238
5,623,312	4/1997	Yan et al.	348/416
5,687,095	11/1997	Haskell et al.	348/394

FOREIGN PATENT DOCUMENTS

0 651 574	5/1995	European Pat. Off. .
0 719 055	6/1996	European Pat. Off. .
23536	10/1994	WIPO .
WO 95/19072	7/1995	WIPO .

OTHER PUBLICATIONS

VBR Transport of CBR-encoded video over ATM Networks, Mei Yong et al., D18.1–D18.3, 1994.

Reduction of the Bit Rate of Compressed Video while in its Coded Form, Morrison et al., D17.1–D17.4, 1994.

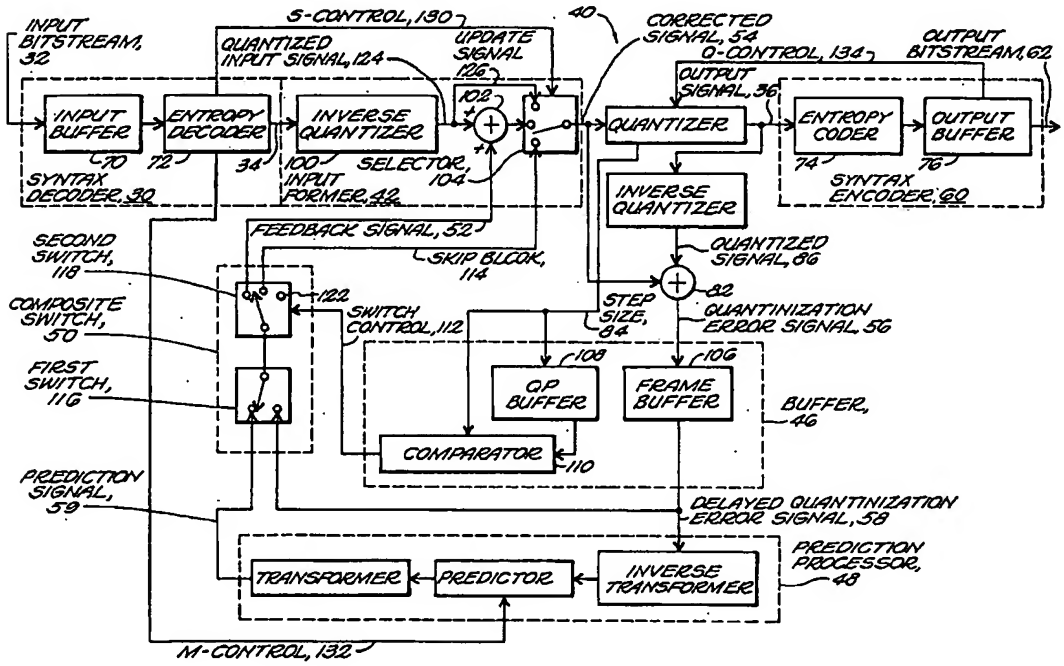
Line Transmission of Non-Telephone Signals. ITU-T H.261, Mar. 1993.

Primary Examiner—Richard Lee
Assistant Examiner—Nhon T. Diep
Attorney, Agent, or Firm—Lahive & Cockfield, LLP

[57] **ABSTRACT**

A transcoding method and apparatus having an input former, a requantizer, a buffer, a prediction processor, and a composite switch. The transcoder has multiple feedback paths routed through the composite switch that are selected depending upon system requirements. The input former can further include an inverse quantizer, a summer, and a selector. The selector connects different signal paths in the transcoder depending upon the coding mode of the input bitstream. The buffer can include a frame buffer, a QP buffer, and a comparator for generating a control signal. Further features of the transcoding method and apparatus include a frequency weighted thresholder and a computation resource manager.

37 Claims, 11 Drawing Sheets



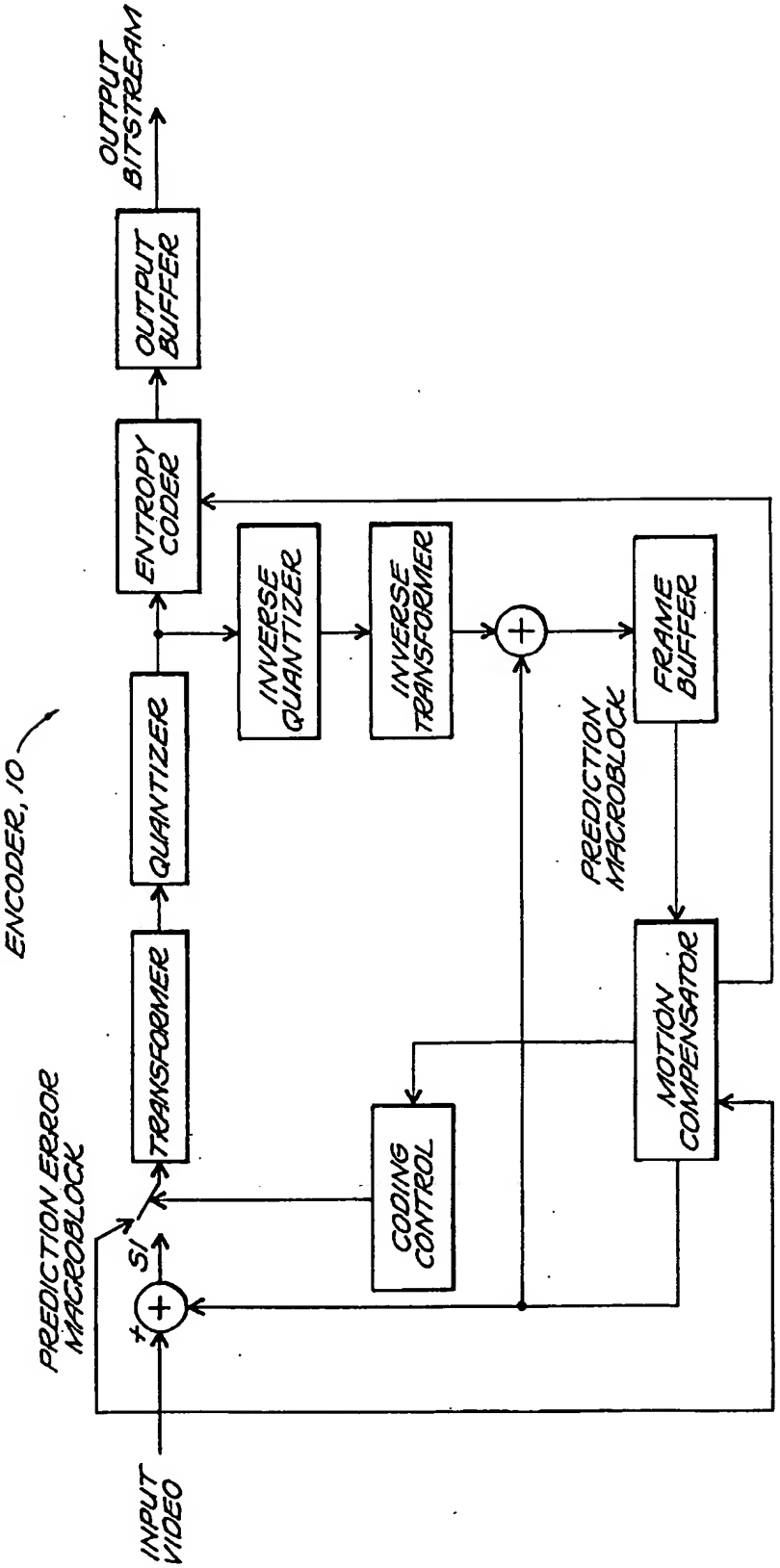


FIG. 1
(PRIOR ART)

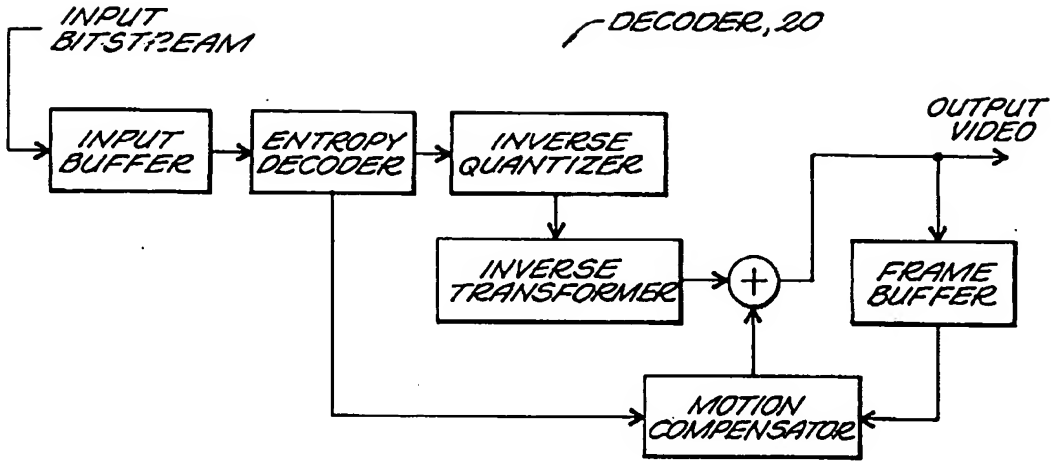


FIG. 2
(PRIOR ART)

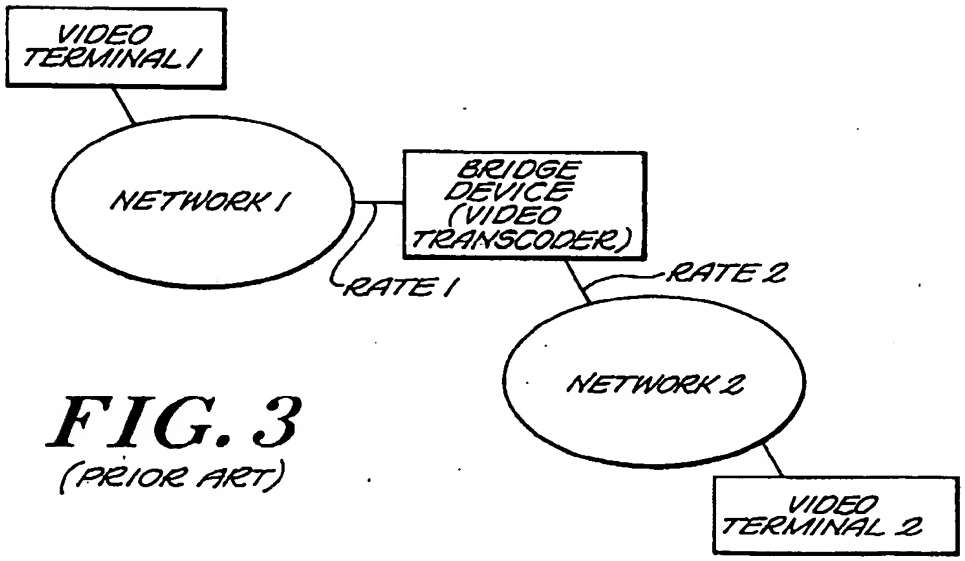


FIG. 3
(PRIOR ART)

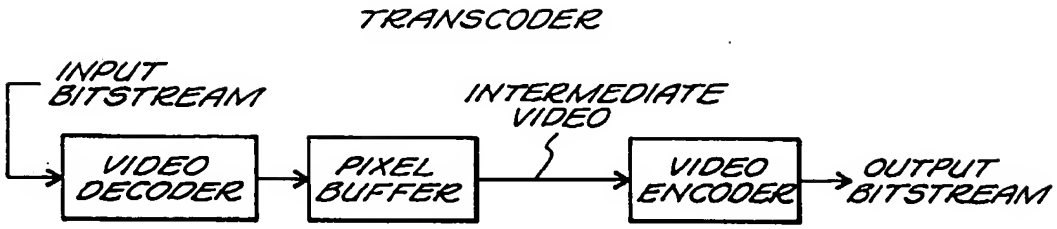


FIG. 4
(PRIOR ART)

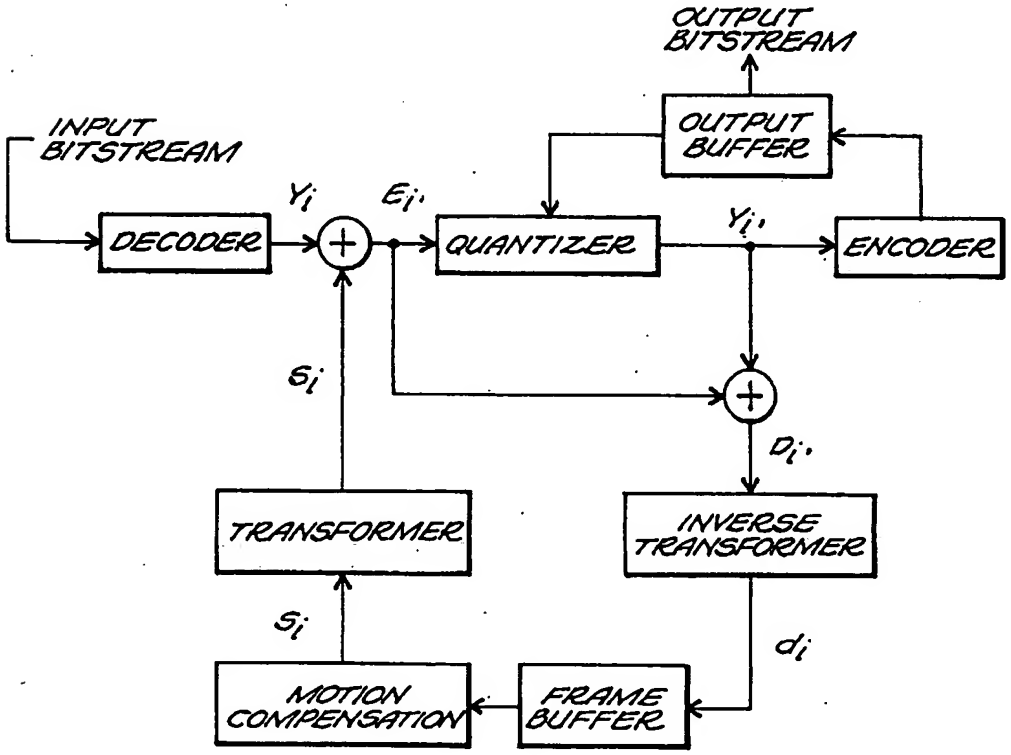


FIG. 5
(PRIOR ART)

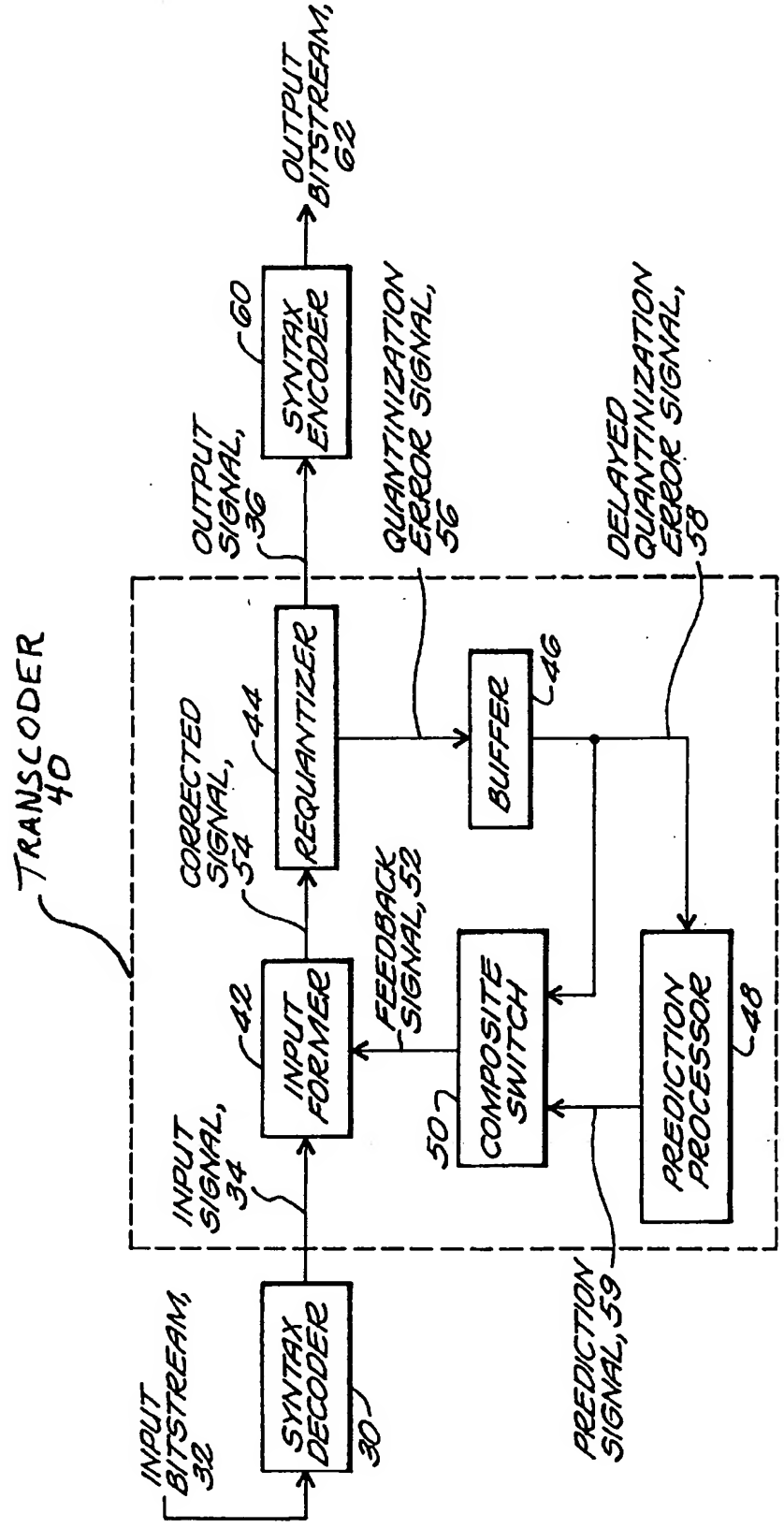

TRANSCODER
40
INPUT BITSTREAM, 32
SYNTAX DECODER
30
INPUT SIGNAL, 34
INPUT FORMER
42
CORRECTED SIGNAL, 54
REQUANTIZER
44
OUTPUT SIGNAL, 36
SYNTAX ENCODER
60
OUTPUT BITSTREAM, 62
FEEDBACK SIGNAL, 52
COMPOSITE SWITCH
50
BUFFER
46
QUANTINIZATION ERROR SIGNAL, 56
DELAYED QUANTINIZATION ERROR SIGNAL, 58
PREDICTION PROCESSOR
48
PREDICTION SIGNAL, 59

FIG. 6

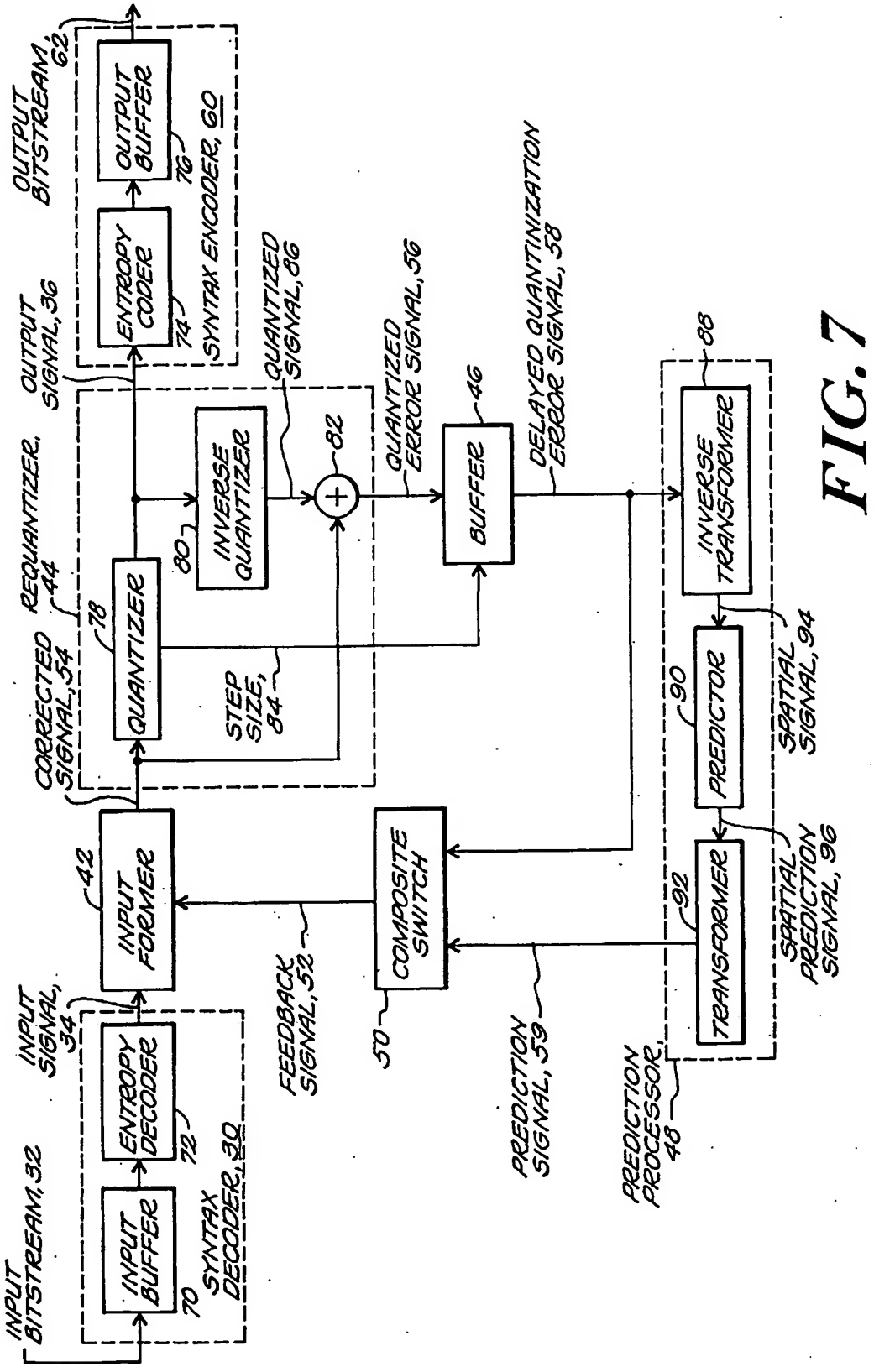

INPUT BITSTREAM, 32
INPUT BUFFER 70
ENTROPY DECODER 72
SYNTAX DECODER, 30
INPUT SIGNAL, 34
INPUT FORMER 42
FEEDBACK SIGNAL, 52
COMPOSITE SWITCH 50
PREDICTION SIGNAL, 59
PREDICTION PROCESSOR, 48
TRANSFORMER 92
SPATIAL PREDICTION SIGNAL, 96
PREDICTOR 90
SPATIAL SIGNAL, 94
INVERSE TRANSFORMER 88
DELAYED QUANTINIZATION ERROR SIGNAL, 58
BUFFER 46
QUANTIZED ERROR SIGNAL, 56
82
INVERSE QUANTIZER 80
QUANTIZED SIGNAL, 86
REQUANTIZER, 44
QUANTIZER 78
CORRECTED SIGNAL, 54
STEP SIZE, 84
OUTPUT SIGNAL, 36
ENTROPY CODER 74
SYNTAX ENCODER, 60
OUTPUT BUFFER 76
OUTPUT BITSTREAM, 62

FIG. 7

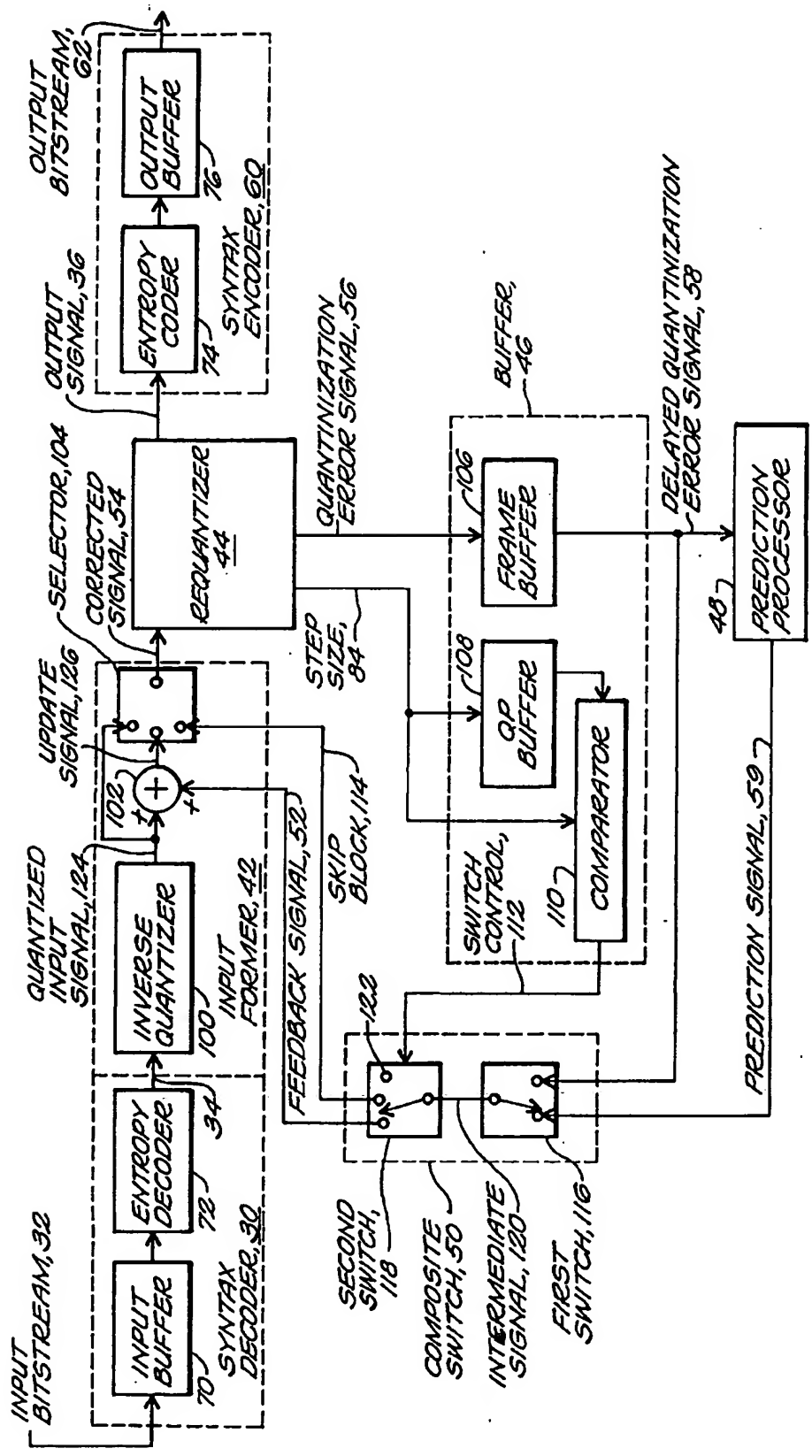

INPUT BITSTREAM, 32
INPUT BUFFER
70
SYNTAX DECODER, 30
ENTROPY DECODER
72
34
INVERSE QUANTIZER
100
INPUT FORMER, 42
QUANTIZED INPUT SIGNAL, 124
102
UPDATE SIGNAL, 126
SELECTOR, 104
CORRECTED SIGNAL, 54
DEQUANTIZER 44
STEP SIZE, 84
QUANTINIZATION ERROR SIGNAL, 56
ENTROPY CODER
74
SYNTAX ENCODER, 60
OUTPUT SIGNAL, 36
OUTPUT BUFFER
76
OUTPUT BITSTREAM, 62
FEEDBACK SIGNAL, 52
SKIP BLOCK, 114
BUFFER, 46
FRAME BUFFER
106
QP BUFFER
108
SWITCH CONTROL, 112
COMPARATOR
110
DELAYED QUANTINIZATION ERROR SIGNAL, 58
PREDICTION PROCESSOR
48
PREDICTION SIGNAL, 59
122
SECOND SWITCH, 118
COMPOSITE SWITCH, 50
INTERMEDIATE SIGNAL, 120
FIRST SWITCH, 116

FIG. 8

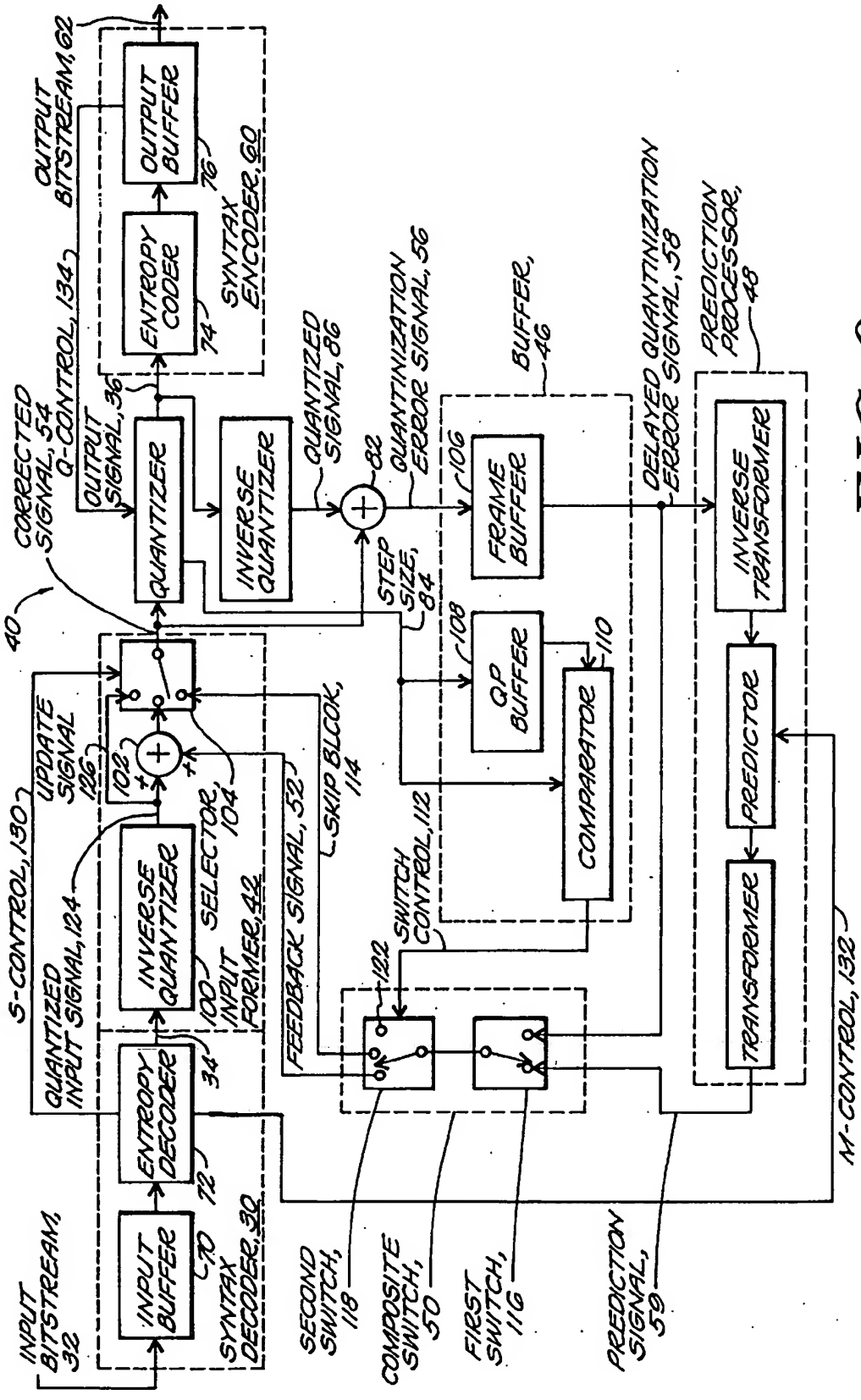

INPUT BITSTREAM, 32
INPUT BUFFER, 70
ENTROPY DECODER, 72
SYNTAX DECODER, 30
QUANTIZED INPUT SIGNAL, 124
S-CONTROL, 130
UPDATE SIGNAL, 126
INVERSE QUANTIZER, 100
SELECTOR, 104
INPUT FORMER, 42
FEEDBACK SIGNAL, 52
SKIP BLOCK, 114
SECOND SWITCH, 118
COMPOSITE SWITCH, 50
FIRST SWITCH, 116
PREDICTION SIGNAL, 59
SWITCH CONTROL, 112
COMPARATOR, 110
QP BUFFER, 108
FRAME BUFFER, 106
BUFFER, 46
INVERSE TRANSFORMER, 102
PREDICTOR, 104
TRANSFORMER, 44
PREDICTION PROCESSOR, 48
DELAYED QUANTINIZATION ERROR SIGNAL, 58
M-CONTROL, 132
STEP SIZE, 84
QUANTINIZATION ERROR SIGNAL, 56
QUANTIZED SIGNAL, 86
INVERSE QUANTIZER
QUANTIZER
OUTPUT SIGNAL, 36
CORRECTED SIGNAL, 54
Q-CONTROL, 134
ENTROPY CODER, 74
OUTPUT BUFFER, 76
SYNTAX ENCODER, 60
OUTPUT BITSTREAM, 62
40

FIG. 9

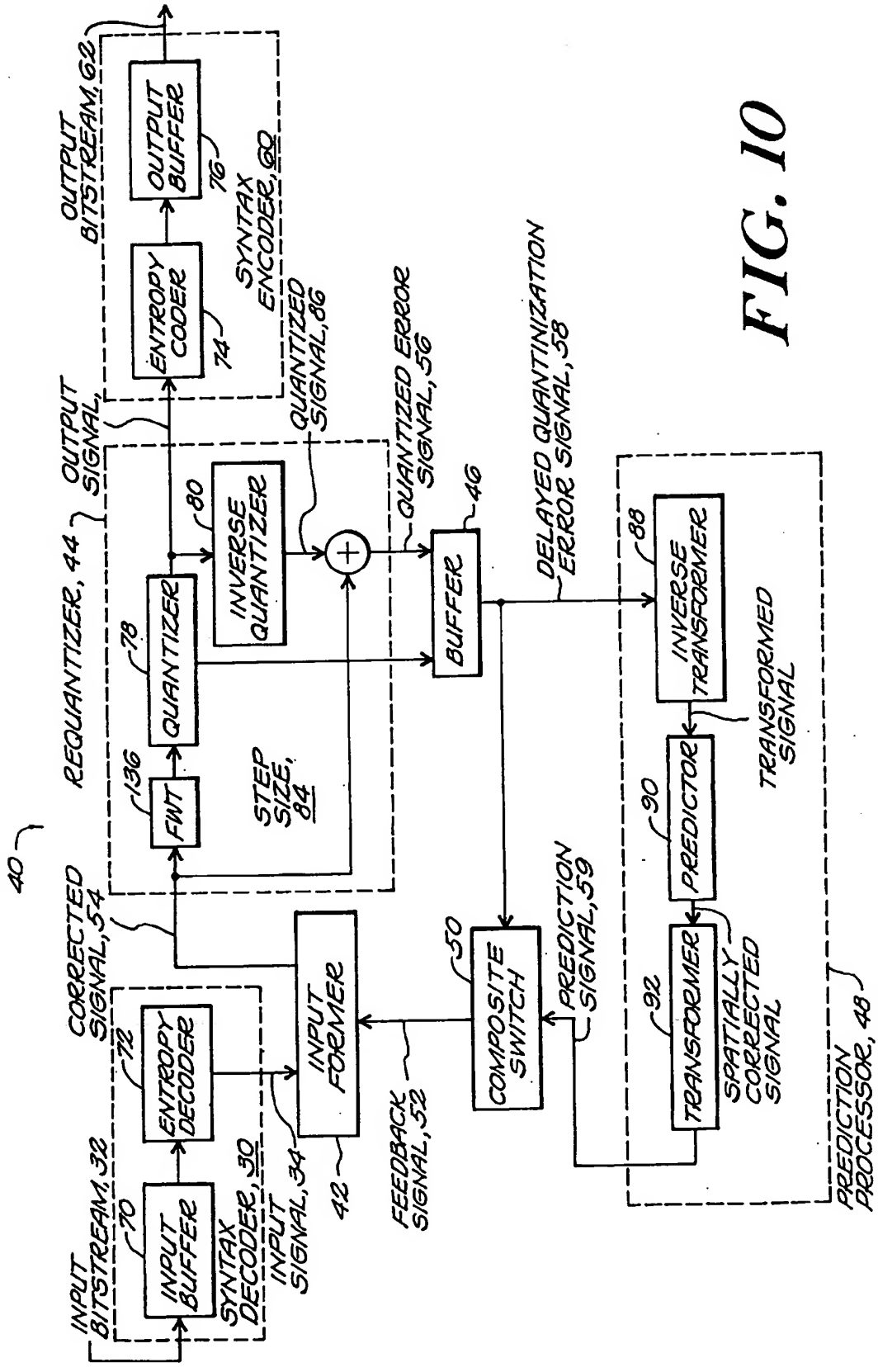

40
REQUANTIZER, 44
OUTPUT SIGNAL,
CORRECTED SIGNAL, 54
INPUT BITSTREAM, 32
INPUT BUFFER
70
SYNTAX DECODER, 30
ENTROPY DECODER
72
INPUT SIGNAL, 34
INPUT FORMER
42
FEEDBACK SIGNAL, 52
136
FWT
78
QUANTIZER
80
INVERSE QUANTIZER
STEP SIZE, 84
QUANTIZED SIGNAL, 86
QUANTIZED ERROR SIGNAL, 56
BUFFER
46
DELAYED QUANTINIZATION ERROR SIGNAL, 58
COMPOSITE SWITCH
50
PREDICTION SIGNAL, 59
88
INVERSE TRANSFORMER
TRANSFORMED SIGNAL
90
PREDICTOR
92
TRANSFORMER
SPATIALLY CORRECTED SIGNAL
PREDICTION PROCESSOR, 48
OUTPUT BITSTREAM, 62
OUTPUT BUFFER
76
ENTROPY CODER
74
SYNTAX ENCODER, 60

FIG. 10

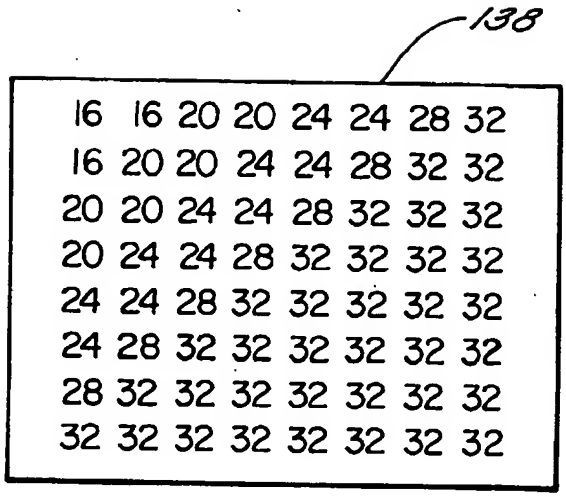

138
16 16 20 20 24 24 28 32
16 20 20 24 24 28 32 32
20 20 24 24 28 32 32 32
20 24 24 28 32 32 32 32
24 24 28 32 32 32 32 32
24 28 32 32 32 32 32 32
28 32 32 32 32 32 32 32
32 32 32 32 32 32 32 32

FIG. 11

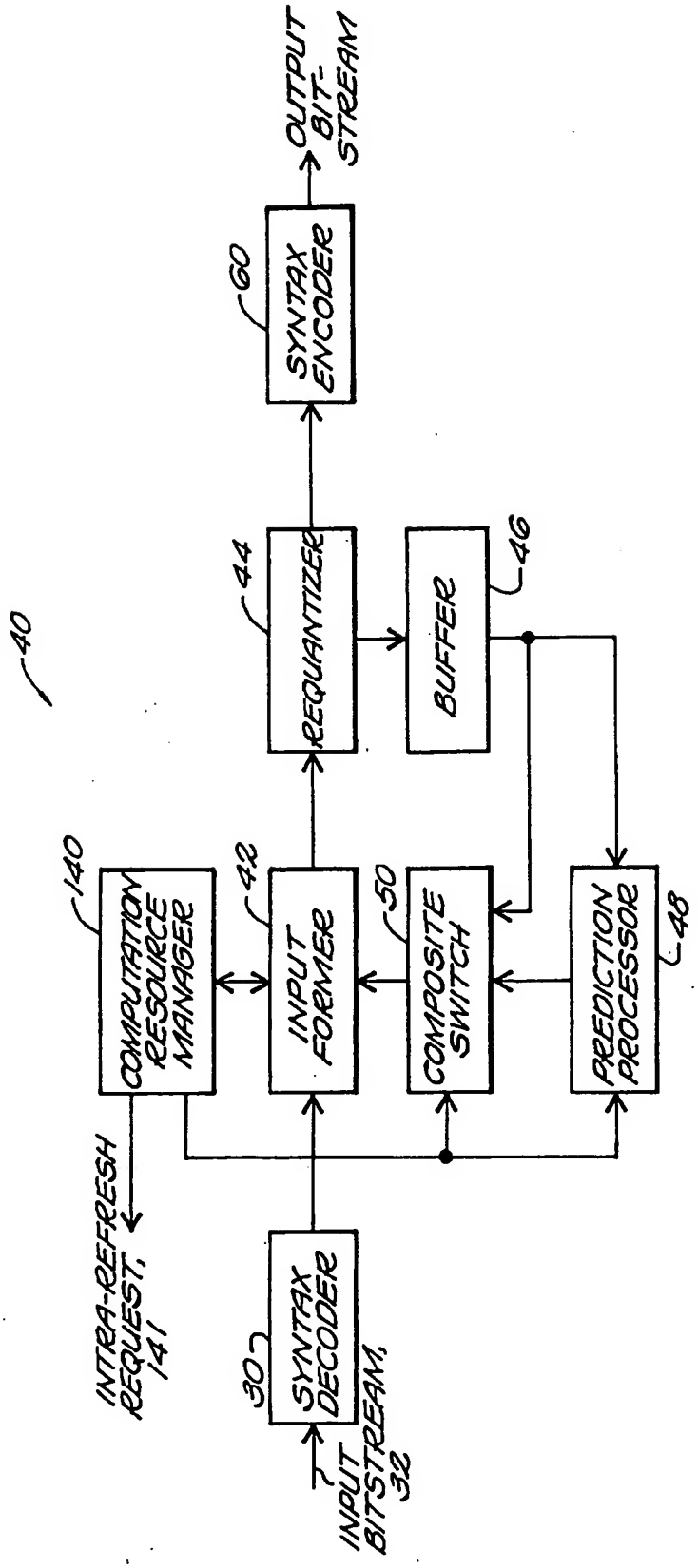

40
140
COMPUTATION RESOURCE MANAGER
42
INPUT FORMER
44
REQUANTIZER
46
BUFFER
48
PREDICTION PROCESSOR
50
COMPOSITE SWITCH
60
SYNTAX ENCODER
OUTPUT BIT-STREAM
30
SYNTAX DECODER
INPUT BITSTREAM, 32
INTRA-REFRESH REQUEST, 141

FIG. 12

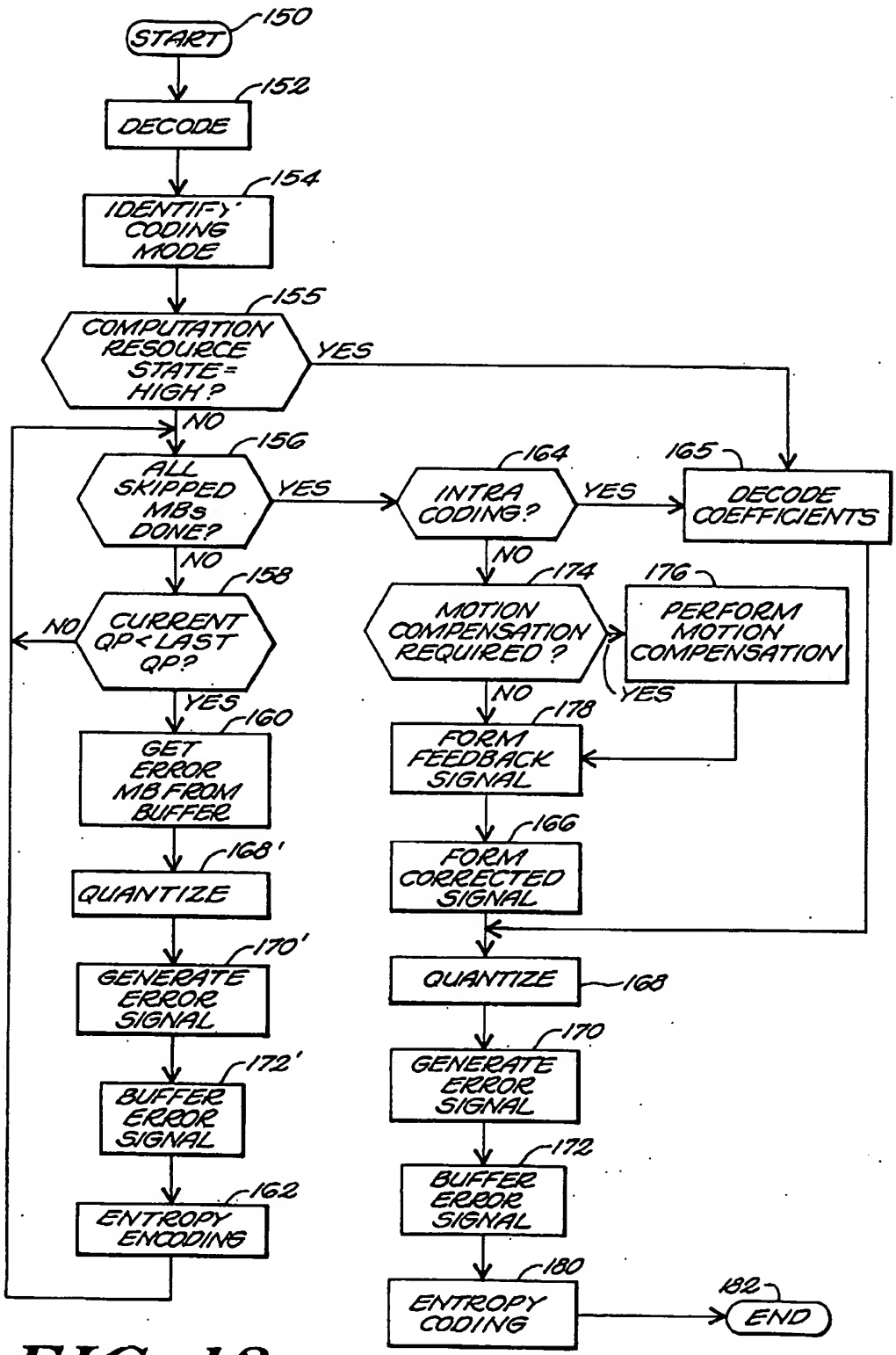

START
150
DECODE
152
IDENTIFY CODING MODE
154
COMPUTATION RESOURCE STATE = HIGH?
155
YES
NO
ALL SKIPPED MBs DONE?
156
YES
NO
CURRENT QP < LAST QP?
158
NO
YES
GET ERROR MB FROM BUFFER
160
QUANTIZE
168'
GENERATE ERROR SIGNAL
170'
BUFFER ERROR SIGNAL
172'
ENTROPY ENCODING
162
INTRA CODING?
164
YES
NO
DECODE COEFFICIENTS
165
MOTION COMPENSATION REQUIRED?
174
PERFORM MOTION COMPENSATION
176
YES
NO
FORM FEEDBACK SIGNAL
178
FORM CORRECTED SIGNAL
166
QUANTIZE
168
GENERATE ERROR SIGNAL
170
BUFFER ERROR SIGNAL
172
ENTROPY CODING
180
END
182

FIG. 13

DEVICE AND METHOD FOR DIGITAL VIDEO TRANSCODING

BACKGROUND OF THE INVENTION

This invention relates to transcoders, and particularly to digital video transcoders for real time conversion between a first and a second coding scheme.

Due to the fast advances in digital technology and VLSI (very large scale integration) technology and the acceptance of international standards, digital video now finds applications in many areas. For example, multimedia, videoconferencing and videotelephony applications all utilize digital video. In each of these applications, occasions exist where data representing moving picture television or sound must be transmitted over long distances via a transmission link. Transmitting the data in digital form, however, requires high bandwidth communication channels and is expensive. To overcome these disadvantages, various techniques have been designed to compress the digitized video data and to reduce the bit rate required to transmit the coded video signals.

Video compression techniques reduce the amount of data being transmitted by allowing for an acceptable degree of degradation in picture quality. Possible correlations that can be exploited to compress the digital data include the spatial correlations among neighboring pixels in an image and the temporal correlation between successive images. For instance, transform techniques reduce the amount of information needed to code a particular frame by removing the statistical redundancy among neighboring samples using prediction methods. One known form of data compression employs predictive coders where an original sample is predicted based upon past samples and a prediction error (the difference between the original and the predicted samples). Predictive coders can be used for speech, image, or video compression.

The state-of-the-art for video compression is the hybrid coding method where predictive encoding is used in reducing temporal redundancy and transform coding is used in eliminating spatial correlation. The ITU-T, H.261 and H.263 recommendations employ this hybrid coding technique to achieve data compression. The H.261 and H.263 guidelines suggest utilizing motion-compensated prediction for temporal prediction, and discrete cosine transforms (DCT) for spatial domain processing. The coded data stream can be reconstituted into a series of video signals by a compatible decoder which decompresses the coded data stream.

FIG. **1** illustrates a block diagram of a hybrid video encoder **10**. According to the ITU-T recommendations, the video input and output consists of individual frames coded in either Common Intermediate Format (hereinafter "CIF") or Quarter Common Intermediate Format (hereinafter "QCIF"). Each frame of the video signal has one luminance component (Y) and two chrominance components (Cb and Cr). Typically, the chrominance components have only half of the resolution of the their luminance counterpart. For example, for a CIF video, the Y component has 288 lines of 352 pixels while the Cb and Cr only have 144 lines of 176 pixels.

To encode a video frame, the frame is first segmented into non-overlapping square blocks. Each of the non-overlapping square blocks is called a macroblock (hereinafter "MB"), and the size of the macroblocks is N×N for luminance components and is (N/2)×(N/2) for chrominance components, where N is a predetermined integer. A typical value for N is 16.

The encoder of FIG. 1 shows that each MB can be coded either in the intra mode or in the inter mode. In the so-called intra mode, the input MB is directly fed to the transformer by a switch **S1**. Intra mode coding is selected to aid in combating transmission error propagation resulting from predictive systems or when there is no similar part in the previous video frame for prediction. Inter mode coding is normally selected to provide for data compression using predictive coding.

When the encoder codes the input video in the inter mode, the encoder searches for a N×N block of data in the previously decoded frame that minimizes a predetermined error function. This N×N block of data, called the prediction macroblock, is retrieved from a frame buffer by a motion compensator. The encoder then forms a prediction error macroblock by finding the difference between the input MB and the prediction MB at a summer. A transformer then transforms the prediction error macroblock output by the summer. The output of the transformer is then quantized by a quantizer. Typically, an entropy coder completes the encoding of the video data, and an output buffer smoothes out the variation in bit generation to achieve the desired bitrate for the output channel. An inverse quantizer is also coupled with the output of the quantizer. The inverse quantizer and an inverse transformer return the signal to its original format. This allows the outputs of the inverse transformer and the motion compensator to be summed and stored in a frame buffer for coding of the next frame.

Further aspects of inter mode coding provide for skipped Macroblocks. Skipped, or non-coded, Macroblocks are formed when the prediction error is so small that after quantization, all the coefficients are quantized to zeroes and the motion vectors associated with the MB are zero. In H.261 format, the skipped MBs can be coded by differentially coding the macroblock address (MBA). For example, if an MBA of 3 is received for the current non-skipped MB is received, then it suggests that the previous 2 MB are skipped. In H.263, 1 bit is used to signal if the current MB is coded or not. If it is 1, then the current MB is skipped. If it's 0, then the current MB is non-skipped (or coded).

FIG. **2** illustrates a block diagram schematic of a decoder **20** known in the art. It may be viewed as a subset of the corresponding encoder shown in FIG. **1**. An input buffer of the decoder buffers the incoming binary bit stream, and an entropy decoder decodes the coding mode, motion vectors and transform coefficients in the incoming bit stream. An inverse quantizer then inverse quantizes the transform coefficients, and an inverse transformer transforms the quantized coefficients. With the aid and direction of the decoded mode information and motion vector, a prediction loop generates an output video signal from the output of the inverse transformer. The prediction loop includes a summer that adds the output of the inverse transformer and the motion compensator to generate the output video signal, a frame buffer for storing and retrieving the output video signal, and a motion compensator for adjusting the signal from the frame buffer according to the motion vector.

When an encoder-decoder pair are attached to the ends of the same communication channel, such as a network, the encode and decode system accurately transmits the digital data over the communication channel. However, in real applications, there are many situations where the two terminals are attached to different communication channels. Communication channels may have different operating rates or characteristics that complicate the encoding and decoding of the digital data. For example, FIG. **3** illustrate a communication system having a first video terminal connected to

network 1 and operating at rate 1, and a second video terminal connected to network 2 and operating at rate 2. A bridge device, having a transcoder, must be used to allow terminal 1 to talk to terminal 2. The bridge device performs both network protocol conversion and video transcoding. Such a bridge device is commonly referred to as a multipoint control unit (MCU) if more than two terminals are involved in a conference.

Transcoders convert the incoming rate of the bit stream into a rate acceptable to the decoding or output end of the bit stream. Transcoders can be used to change the rate of a digital video bit stream in a network, or to convert one video compression format to another. Additionally, transcoders can convert a constant bit rate stream into a variable bit rate stream to save bandwidth through statistical multiplexing. Another important application of transcoding involves multipoint videoconferencing where transcoders mix video signals together for continuous presence teleconferencing.

FIG. 4 illustrates a transcoder formed by cascading a video decoder with a video encoder. The decoder decodes the input bit stream into pixel data, and the encoder encodes the intermediate video into the output bit stream with the desired algorithm and bitrate. This transcoder introduces processing delay and requires significant implementation complexity.

FIG. 5 is a block diagram of a more efficient video transcoder as disclosed in: (1.) Eyuboglu, U.S. Pat. No. 5,537,440; (2.) M. Yong, Q.-F. Zhu, and V. Eyuboglu, "VBR transport of CBR encoded video over ATM networks," Sixth Int. Workshop on Packet Video, Sept. 1994; and (3.) D. G. Morrison, M. E. Nilsson and M. Ghanbari, "Reduction of the bit-rate of compressed video while in its coded form," Sixth Int. Workshop on Packet Video, Sept. 1994. As shown in FIG. 5, a modified version S_i of the quantinization error vector is subtracted from a received quantized vector Y_i to form a difference vector E_i'. The difference vector E_i' is requantized to obtain the transcoded vector Y_i'. A quantinization error vector calculator then computes the inverse transformed quantinization error vector d_i equal to $A_i^{-1}[D_i']$, where $D_i'=Y_i'-E_i'$ is the quantinization error vector and A_i^{-1} is an inverse transformation. A modifying circuitry determines the modified quantinization error vector S_i based on the past vectors d_i'. The modifying circuitry can include a transformer, a motion compensator, and a frame buffer. The frame buffer stores past values of the vectors d_i'.

The approach illustrated in FIG. 5, however, does not fully account for the processing of skipped macroblocks. This transcoder can allow noise introduced during the transcoding of past frames to remain in the image signal of a skipped macroblock. This effect creates low quality image signals that represent the background in a video signal. Additionally, the transcoder shown in FIG. 5 fails to provide for a feedback path that does not require motion compensation.

Accordingly, it is an object of the invention to provide for a more efficient and cost effective transcoder.

A particular object of the invention is to effectively provide for the processing of skipped macroblocks.

Another object of the invention is to provide a transcoder having a plurality of feedback paths.

These and other objects of the invention are evident in the following description.

SUMMARY OF THE INVENTION

The invention achieves the foregoing objects with a transcoding method and apparatus having an input former that generates a corrected signal based upon an input bitstream of digitized data and a feedback signal, and a requantizer that quantizes the corrected signal to form an output signal having the desired quantinization level. The requantizer also generates an error signal representative of the quantinization error in the output signal. The novel transcoder further includes a buffer for storing and retrieving the quantinization error signal generated by the requantizer, and a prediction processor for generating a prediction signal based upon a delayed quantinization error signal received from the buffer. A composite switch generates and forwards to the input former the feedback signal. The composite switch generates the feedback signal by selecting between the delayed quantinization error signal and the prediction signal. Unlike prior art systems, this transcoder provides for selectable feedback paths, each path being capable of generating a feedback signal. Shorter and less computationally intensive feedback paths rapidly form substantially noiseless feedback signals, while longer and more computationally expensive feedback paths can be utilized when needed to process more complicated image signals.

One aspect of the invention provides for an input former having a selector that modifies the corrected signal based upon the coding mode of the input bitstream.

The selector connects different signal paths depending upon whether the input bitstream is coded as an intra mode, inter mode, or skipped block signal.

Other features of the invention include a buffer having a frame buffer, a QP buffer, and a comparator. The frame buffer stores and retrieves the quantinization error signal generated by the requantizer. The QP buffer and the comparator provide additional control signals to the transcoder for the processing of data encoded in the skipped block mode. In particular, the comparator generates a control signal based upon the current quantinization step size utilized by the requantizer and a past step size stored in the QP buffer.

In accordance with further aspects of the invention, the composite switch can include a first switch and a second switch. The first switch selectively determines an input signal to the composite switch, and the second switch selectively determines an output signal from the composite switch.

Another aspect of the invention includes a frequency weighted thresholder operably coupled with the input former. The frequency weighted thresholder generates a frequency weighted corrected signal by applying frequency dependent threshold limits to the corrected signal. The corrected signal formed generates a video image more suited to the particulars of human vision. An additional feature of the invention is a computation resource manager. The resource manager monitors the computational usage status of the transcoder and controls the computational usage of the transcoder in response to the monitored usage.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram schematic of a prior art hybrid video encoder;

FIG. 2 is a block diagram schematic of a prior art hybrid video decoder;

FIG. 3 illustrates two prior art video networks connected through a bridging device having a transcoder;

FIG. 4 shows a block diagram of a known prior art transcoder;

FIG. 5 shows a block diagram of another prior art transcoder;

FIG. 6 is a block diagram of a transcoder in accordance with the present invention;

FIG. 7 illustrates a further embodiment of the transcoder shown in FIG. 6 having a detailed diagram of a requantizer and a prediction processor;

FIG. 8 shows a block diagram of the transcoder of FIG. 6 having a detailed diagram of an input former and of a predictor;

FIG. 9 is a detailed block diagram of the transcoder of FIG. 6;

FIG. 10 is a detailed block diagram of the transcoder of FIG. 6 having a frequency weighted thresholder;

FIG. 11 is an example of a threshold matrix associated with the frequency weighted thresholder of FIG. 10;

FIG. 12 shows the transcoder of FIG. 6 having a computation resource manager; and

FIG. 13 illustrates a flow chart of the transcoding method in accordance with the present invention.

DETAILED DESCRIPTION OF THE INVENTION

FIG. 6 illustrates a transcoding system having a syntax decoder 30, a transcoder 40 according to the invention, and a syntax encoder 60. The transcoding system can be incorporated into a communication system so that the transcoder 40 acts as a bridge between two networks. The syntax decoder 30 receives an input bit stream 32 from one network, and generates an input signal 34 that is fed to the transcoder 40. When operating within the confines of the H.261 standards, the input signal 34 generated by the syntax decoder 30 is macroblock data and side information from the input bit stream 32 pertaining to framing, inter/intra mode, step size, and motion vectors. The syntax encoder 60 receives an output signal 36 from the transcoder 40 and generates an output bit stream 62 in response thereto.

The transcoder 40 includes an input former 42, a requantizer 44, a buffer 46, a prediction processor 48, and a composite switch 50. The input former 42 receives the input signal 34 from the syntax decoder 30 and a feedback signal 52 from the composite switch 50. The input former 42 generates a corrected signal 54 in response to the received input signal 34 and feedback signal 52. The requantizer 44 quantizes the corrected signal 54 and generates a quantized output signal 36 and a quantinization error signal 56. The buffer receives the quantinization error signal 56 and generates a delayed quantinization error signal 58. The delayed quantinization error signal 58 is routed both to the prediction processor 48 and to the composite switch 50. The composite switch 50 receives a prediction signal 59 from the prediction processor and a delayed quantinization error signal 58 from the buffer. The composite switch 50 generates the feedback signal 52 by selecting between the delayed quantinization error signal 58 and the prediction signal 59.

The transcoder 40 shown in FIG. 6 advantageously includes two quantinization error feedback paths that lead from the requantizer 44 and proceed through the buffer 46 to the input former 42. The shorter feedback path proceeds directly from the buffer 46 to the composite switch 50. This shorter feedback path occurs when the composite switch 50 creates a feedback path that leads from the buffer 46 directly to the input former 42. The longer feedback path proceeds from the buffer 46, through the prediction processor 48, to the composite switch 50. This longer feedback path is created when the composite switch 50 forms a feedback path that proceeds from the prediction processor 48 to the input former 42.

The shorter feedback path generates a feedback signal that is frequently more accurate than a feedback signal generated by the longer feedback path. The longer feedback path through the prediction processor 48 creates additional distortions in the feedback signal 52 because of the additional processing of the delayed quantinization error signal 58 that occurs in the prediction processor 48. The prediction signal 59 is frequently a less accurate representation of the quantinization noise than the delayed quantinization error signal 58, and the prediction signal 59 typically takes longer to generate than the immediate routing action of the delayed quantinization error signal 58. However, the longer feedback path proves to be effective when spatial manipulation or spatial filtering of the feedback signal 52 is necessary. For instance, the longer feedback path proves particularly effective when the signal must undergo motion compensation. Accordingly, the invention provides for a selectable feedback path for generating the feedback signal 52.

In operation, the transcoder 40 of FIG. 6 accepts a binary digital videostream generated by a remote encoder according to a predetermined syntax. The transcoder 40 also outputs a binary digital videostream having a bit rate that may differ from the rate of the original input bit stream 32. The output bit stream 62 can be formed having the same syntax as the original input bit stream 32 or it can be formed having a different syntax than the original input bit stream 32.

The syntax decoder 30 performs a partial decoding of the input bit stream 32 and thereby generates a set of signals, including: coding mode signals, motion vectors, and transform coefficients. The input former 42 receives the transform coefficients from the syntax decoder 30 and a feedback signal 52 from the composite switch 50. The input former 42 generates, in response to the feedback signal 52 and to the transform coefficients, a corrected signal 54. The corrected signal 54 represents an input signal 34 that has been modified by the feedback signal 52. Typically the feedback signal 52 is a measure of the quantinization error introduced by the requantizer 44, and accordingly the corrected signal 54 is updated to counteract the noise introduced by the requantizer 44.

The requantizer 44 quantizes the output of the input former 42, i.e. the corrected signal 54, to generate a quantized output signal 36 having a bit rate that differs from the bit rate of the input bit stream 32. The requantizer 44 also generates a quantinization error signal 56 based upon the difference between the input to the requantizer 44 and the output of the requantizer 44. The syntax encoder 60 receives the quantized output signal 36 from the requantizer 44 and encodes the output of the requantizer 44, together with the motion vectors, to generate the output bit stream 62 according to a predetermined syntax. Typically, the syntax is the same as that of the input bit stream 32.

The buffer 46 receives the quantinization error signal 56 from the requantizer 44. The quantinization error signal 56 is a measure of the error introduced to the corrected signal 54 during quantinization of the corrected signal 54. The buffer 46 generates a delayed quantinization error signal 58 that represents a prediction of the quantinization noise for the next input signal 34. Generally, the delayed quantinization error signal 58 is based upon past values of the quantinization error signal 56.

The prediction processor 48 receives the delayed quantinization error signal 58 from the buffer. The prediction processor 48 then generates a prediction signal 59 in response to the delayed quantinization error signal 58. The

prediction processor **48** modifies the delayed quantinization error signal **58** in the spatial domain, while typically the delayed quantinization error signal **58** is located in another domain, such as the frequency domain. The prediction processor **48** operates within the spatial domain because certain aspects of the delayed quantinization error signal **58** may be more easily manipulated in the spatial domain. After the prediction processor **48** has modified the delayed quantinization error signal **58** within the spatial domain, the prediction processor **48** transforms the modified delayed quantinization error signal **58** back to its original domain and outputs the signal as the prediction signal **59**.

The composite switch **50** receives the delayed quantinization error signal **58** from the buffer **46** and the prediction signal **59** from the prediction processor **48**. The output of the composite switch **50** is coupled with the input former **42** and is designated as a feedback signal **52**. The composite switch **50** is controllable to either create a signal path between the buffer **46** and the input former **42** (i.e. the shorter feedback path), or to create a signal path between the prediction processor **48** and the input former **42** (i.e. the longer feedback path). When the composite switch **50** selects the shorter feedback path the feedback signal **52** depends directly upon the delayed quantinization error signal **58**, and when the composite switch **50** selects the longer feedback path the feedback signal **52** depends upon the prediction signal **59**.

FIG. **7** illustrates further aspects of the transcoder **40** in accordance with the invention. In particular, FIG. **7** shows an expanded view of the syntax decoder **30**, the syntax encoder **60**, the requantizer **44**, and the prediction processor **48**.

The syntax decoder **30** includes an input buffer **70** and an entropy decoder **72**. The input buffer **70** stores and retrieves the input bit stream **32** and the entropy decoder **72** decodes the input bit stream **32** into transform coefficients and side information. The transform coefficients are forwarded to the input former **42**. The side information allows the syntax decoder **30** to correctly reconstruct the coded signal. Side information can include the source format (CIF or QCIF), quantizer **78** step size, inter/intra mode decision, and motion vectors. The encoder can also transmit side information to allow the transcoder **40** to skip a macroblock. For example, when all the coefficients in the block are zero and when there is little motion, the macroblock may be identified as a skipped macroblock in the side information.

The syntax encoder **60** includes an entropy coder **74** and an output buffer **76**. The output signal **36** of the requantizer **44** is received by the entropy coder **74**. The entropy coder **74** re-encodes the output signal from the requantizer **44** and forwards the re-encoded binary bits to the output buffer **76**. The output buffer **76** stores the binary bit stream for transmission along a communication channel in the form of an output bit stream **62**.

The transcoder **40** of FIG. **7** also includes a detailed block diagram of the requantizer **44**. The requantizer **44** includes a quantizer **78**, an inverse quantizer **80**, and a summer **82**. The quantizer **78** receives a corrected signal **54** and generates the output signal **36** in response thereto. Quantizers are known in the art and are used to create a discrete level representation of an input signal in accordance with a quantinization step size. Typically, the quantinization step size can be adjusted with a control signal. As the magnitude of the quantinization step size increases, the fineness of the quantinization decreases, and vice-versa. It is further worth noting that quantizers suffer from error. This error, called quantinization error, is the difference between the input value and the reconstruction value of the input signal.

As shown in FIG. **7**, the quantizer **78** is operably coupled to receive the corrected signal **54** from the input former **42**. The quantizer **78** quantizes the coefficients in the corrected signal **54** with step size Delta, where the step size is determined as a function of the difference between the bit rate of the input bit stream **32** and the desired bit rate of the output bit stream **62**. The step size is further determined as a function of the desired image quality. The step size is determined so as to achieve the desired change in bit rate while improving or retaining image quality. Accordingly, the bit rate of a bit stream can be changed using the transcoder **40** of the present invention.

The current quantinization step is particularly important during the skipped block mode. Data regarding the current quantinization step size aids in determining how to reduce noise in the output signal **36** during the skipped block mode of operation. Accordingly, a step size signal **84**, representing the quantinization step size currently being utilized by the quantizer is output by the quantizer. As shown in FIG. **7**, the buffer **46** receives the step size signal **84**.

The requantizer **44** also includes an inverse quantizer **80** and a summer coupled with the quantizer **78** for generating a quantized error signal **56** representative of the noise added to the corrected signal **54** by the quantinization process. In particular, the inverse quantizer **80** is operably coupled to receive the output signal **36** from the quantizer **78**. The inverse quantizer **80** performs the corresponding inverse quantinization with the same step size. The summer is operably coupled to receive the corrected signal **54** and is operably coupled with the inverse quantizer **80** to receive the quantized signal **86**. The summer generates a quantized error signal equal to the difference between the corrected signal **54** and the quantized signal **86**. The signal produced by the summer represents the quantinization error generated during the quantinization process.

The prediction processor **48** shown in FIG. **7** includes an inverse transformer **88**, a predictor **90**, and a transformer **92**. The inverse transformer **88** receives the delayed quantinization error signal **58** from the buffer **46** and generates a spatial signal **94**. The inverse transformer **88** transforms the delayed quantinization error signal **58** to a domain more appropriate for spatial modification. The predictor **90** receives the spatial signal **94** and generates a spatial prediction signal **96** that is output to the transformer **92**. The transformer **92** then operates upon the spatial prediction signal **96** with a transform that is the inverse of the inverse transformer **92**. That is, the transformer **92** returns the signal back to the domain of the delayed quantinization error signal **58**.

The transformer **92** used in the transcoder **40** is preferably a discrete cosine transform (DCT) and the inverse transformer **92** is preferably an inverse discrete cosine transform. The inverse discrete cosine transform is applied independently to each 8×8 block in the macroblock to obtain the transform coefficients as given by:

spatial signal 94 $(x, y) =$

$$\frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7} C(u)C(v)F(u, v)\cos[\pi(2x+1)u/16]\cos[\pi(2y+1)v/16]$$

with u,v,x,y=0,1,2, . . . , 7;

where x,y=spatial coordinates in the pel domain;

u,v=coordinates in the transformed domain;

$$C(u) = \frac{1}{\sqrt{2}} \quad \text{;for } u = 0$$

$$1 \quad \text{;for all other cases; and}$$

$$C(v) = \frac{1}{\sqrt{2}} \quad \text{;for } v = 0$$

$$1 \quad \text{;for all other cases;}$$

$F(u, v)$ = delayed quantinization error signal.

The predictor **90** is preferably a motion compensator. The motion compensator adjusts the spatial signal **94** to account for movement in the image being transmitted. The motion compensator relies upon a motion vector that is extracted from the input bitstream by the entropy decoder **72** to implement motion compensation. Typically a motion vector is generated, in a remote encoder, by an algorithm that searches the pixels in a previously reconstructed image to find a window for which the "distance" between the pixels in that window and the corresponding pixels in the current block is a minimum. The spatial offset between the window and the current block is defined as the motion vector. The motion compensator then uses the motion vector to offset the current image signal to obtain a more accurate data representation of the image.

FIG. **8** illustrates further details of the transcoder **40** shown in FIG. **6**. The illustrated input former **42** includes an inverse quantizer **100**, a summer **102** and a selector **104**. In addition, further details of the buffer **46** as shown include a frame buffer **106**, a QP buffer **108** and a comparator **110**.

The buffer **46** shown in FIG. **8** receives a quantinization error signal **56** and a step size signal **84** from the requantizer **44** and generates a delayed quantinization error signal **58** and a switch control signal **112** as output. The quantinization error signal **56** is routed to the frame buffer **106**. The frame buffer **106** generates the delayed quantinization error signal **58** based upon past values of the quantinization error signal **56**. For instance, the delayed quantinization error signal **58** generated can be the quantinization error signal **56** during previous frames. In accordance with this embodiment, the coefficient frame buffer **106** can simply be a memory element for storing and retrieving the quantinization error signal **56**.

The buffer **46** also includes a QP buffer **108** for receiving the step size signal **84** from the requantizer **44**. The QP buffer **108** is typically a memory element large enough for storing and retrieving the step size signal **84** for a video frame. The memory size of the QP buffer **108** is determined by the number of macroblocks in a videoframe multiplied by the space required to store the step size signal **84**. For a typical video coding method like H.261 and H.263, five bits is enough to store the step size signal **84** because the step size signal **84** only takes integer values from one to 31. For a QCIF video with frame size of 144×176, we have 99 macroblocks. Therefore, the minimum QP buffer **108** size for this case is 495 bits. However, it should be noted, that in many practical applications one byte (8 bits) is used to store the step size signal **84** in order to save implementation complexity.

The buffer **46** also can include a comparator **110** for comparing the step size signal with the output of the QP buffer **108**, i.e. a delayed step size signal. The comparator **110** generates a switch control signal **112** in response to the inputs received from the QP buffer **108** and the requantizer **44** and in response to the current operating mode of the transcoder **40**. When the transcoder **40** is not operating in skipped block mode, the control signal **112** directs the composite switch **50** to generate a feedback signal **52**. When the transcoder **40** is operating in skipped block mode, the control signal directs the composite switch **50** to generate a skip block signal **114**.

During skipped block mode, the comparator **110** compares the current step size signal **84** with the delayed step size signal from the QP buffer **108**. If the current step size is less than the delayed step size, then the control signal directs the composite switch **50** to generate a skipped block signal based upon the delayed quantinization error signal **58**. If, however, the current step size is more than the delayed step size, then the control signal directs the composite switch **50** to generate a skipped block signal having a null value.

The transcoder **40** shown in FIG. **8** also includes a detailed diagram of the composite switch **50**. The composite switch **50** has a first switch **116** and a second switch **118**. The first switch **116** controls the selection of an input signal into the composite switch **50** and the second switch **118** controls the selection of an output signal from the composite switch **50**. An intermediate signal path **120** connects the first switch **116** and the second switch **118**.

The first switch **116** selects an input signal to the switch from either the delayed quantinization error signal **58** or the prediction signal **59**. Thus, the intermediate signal is based upon either the delayed quantinization error signal **58** or the prediction signal **59**, depending upon the input terminal chosen. The second switch **118** selects an output signal from either the feedback signal **52**, the skip block signal **114**, or a null terminal **122**. The intermediate signal is routed to either the feedback signal **52**, the skip block signal **114**, or the null position, depending upon the output terminal chosen.

The switch control signal **112** from the buffer **46** control the selection of the output terminal in the second switch **118**. If the transcoder **40** is not operating in skipped block mode, then the intermediate signal is connected with the feedback signal **52**. If the transcoder **40** is operating in skipped block mode, then the intermediate signal is connected with either the skip block signal **114** or the null terminal **122**. Accordingly, when operating in skip block mode, the intermediate signal can be connected to the skip block signal, thereby forming a corrected signal **54** substantially equivalent to the intermediate signal. Alternatively, when operating in skip block mode, the intermediate signal can be connected with the null terminal **122**, thereby forming a corrected signal **54** having a null value.

FIG. **8** also details elements forming the input former **42**. The input former **42** includes an inverse quantizer **100**, a summer **102** and a selector **104**. The inverse quantizer **100** receives an input signal **34** from the syntax decoder **30** and generates a quantized input signal **124**. The summer receives the quantized input signal **124** from the inverse quantizer **100** and a feedback signal **52** from the composite switch **50**. The summer **102** generates an update signal **126** equal to the sum of the input signals it receives. The selector **104** receives the quantized input signal **124**, the update signal **126**, and the skip block signal **114** and selects one of the three received signals to be the corrected signal **54**.

The selector **104** provides different paths for generating the corrected signal **54** to account for various modes of coding the image in the input bit stream **32**. The coding modes include intra mode, inter mode, and skip block mode. For instance, in intra mode coding the input bit stream is based on a single video image, rather than a comparison between frames of the input picture. In this case, the selector **104** forms the corrected signal **54** based solely upon the quantized input signal **124**. When the original macroblock is coded in inter mode, the input bit stream is based upon a

comparison between video frames. Under these circumstances, the selector **104** must couple the corrected signal **54** with the feedback signal **52**. Particularly, in inter mode, the corrected signal **54** is generated by routing the sum of the quantized input signal **124** and the feedback signal **52** through the selector **104**. If the original macroblock coded in the input bit stream **32** is coded as a skipped macroblock, then the selector **104** bases the corrected signal **54** upon the skip block signal **114** received from the composite switch **50**.

FIG. **9** is a detailed schematic of the transcoder **40** and various control signals. A S-control signal **130** and a M-control signal **132** are generated when the entropy decoder **72** decodes the side information from the input bit stream **32**. A Q-control signal **134** is generated by the output buffer **76** and is input to the quantizer **78**.

The Q-control signal **134** is used to control the quantinization step size of the quantizer **78**. The Q-control signal **134** varies as a function of the fullness of the output buffer **76** memory. Accordingly, the quantizer **78** is forced to be responsive to the fullness of the output buffer **76**. For instance, the quantinization step size for a group of data can be determined based upon the quantity of codes stored in the output buffer **76** at the time after the completion of the coding of the preceding group of data. The Q-control signal **134** prevents buffer overflows and underflows by decreasing the quantinization step size. Decreasing the quantinization step size increases the fineness of the quantinization thereby increasing the amount of data in the output signal **36**. Similarly, increasing the quantinization step size decreases the fineness of the quantinization thereby decreasing the amount of data in the output signal **36**.

The S-control signal **130** is produced at the entropy decoder **72** and is used to control the selector **104**. In particular, the S-control signal **130** determines the routing path to form the corrected signal **54**. If the signal decoded at the decoder indicates that the macroblock is coded in intra mode, then the S-control signal **130** causes the selector **104** to route quantized input signal **124** through the selector **104**. If the macroblock is coded as inter mode, then the S-control signal **130** causes the selector **104** to route the update signal **126** through the selector **104**. In addition, if the original macroblock is coded as a skip block, then the S-control signal **130** causes the selector **104** to route the skip block signal **114** through the selector **104**.

FIG. **9** also shows an M-control signal **132** that is generated at the decoder stage from the input bit stream **32**. The M-control signal **132** directs the operation of the composite switch **50** and the motion compensator depending upon the encoding mode of the video image in the input bit stream **32**. If the decoded input bit stream **32** further indicates that only interframe coding is used without motion compensation, then the M-control signal **132** directs the composite switch **50** to connect the path between the delayed quantinization error signal **58** and the feedback signal **52**. In comparison, if the decoded input bit stream **32** indicates that motion compensation is used, then the M-control signal **132** directs the composite switch **50** to connect the path between the prediction signal **59** and the feedback signal **52**.

For relatively low bit rate digital video applications such as videoconferencing and videotelephony, the majority of the macroblocks will be coded without motion compensation. For these macroblocks, the present invention directly fetches their delayed quantinization error signal **58** from the buffer **46** without going through the prediction processor **48**. The advantage of this is two-fold: (1) implementation complexity is significantly reduced, and (2) the additional distortion added to the signal by the prediction processor **48** is eliminated. In particular, the delayed quantinization error signal **58** contains less additive noise than the prediction signal **59** and accordingly the delayed quantinization error signal **58** produces a higher quality picture.

FIG. **10** illustrates the transcoder **40** having a frequency weighted thresholder (FWT) **136**. The FWT **136** is operably coupled to the input former **42**, particularly the FWT **136** receives the corrected signal **54** from the input former **42** and generates a signal that is received by the quantizer **78** in the requantizer **44**. The FWT **136** improves the picture quality of the output signal **36** generated by the requantizer **44**.

It is well known that human vision systems can tolerate higher distortions for higher frequency components than for low frequency components. However, in most video coding systems, such as H.261 and H.263, all the transform coefficients are treated equally. The FWT **136** takes advantage of the particulars of human vision to improve picture quality by applying different weights to the various frequency coefficients of the video image.

The FWT **136** is a filter that is frequency-dependent. The FWT **136** includes thresholds assigned to various frequency components across the spectrum. The FWT **136** compares the input transform coefficients to the thresholds at a given frequency component. If the input transforms coefficients do not exceed the threshold at the given frequency component, then the transform coefficient is set to zero. If the incoming transform coefficient exceeds the threshold value at the given frequency, then the transform coefficient is left unmodified. The threshold values differ across the range of frequency being viewed by the FWT **136**. In particular, the thresholds are higher for higher frequency components than they are for low frequency components.

FIG. **11** illustrates an example of a threshold matrix **138** forming the FWT **136** shown in FIG. **10**. The matrix shown in FIG. **11** is for use with a DCT transform having an 8×8 block size. The real threshold of the FWT **136** are derived as follows:

$$T_{ij}=W_{ij}*QP/8$$

where i=0,1, . . . ,7; and

j=0,1, . . . ,7,

In the above equation i is the row index and j is the column index in the DCT transform domain, T_{ij} is the threshold to be used for coefficients at location (i,j), W_{ij} is the corresponding weighting factor shown in the matrix of FIG. **11**, and QP is half of the quantinization step size used for quantizing the current macroblock for data.

FIG. **12** illustrates a further aspect of the invention for increasing the efficiency of the transcoder **40**. The transcoder **40** shown includes a computation resource manager **140** operably coupled with the syntax decoder **30** and with the input former **42** for monitoring the system load and for controlling the transcoder **40** in response to the system load. The computation resource manager **140** can also be coupled with the composite switch **50** and the prediction processor **48** for controlling computational usage by the transcoder **40**. The computation resource manager is coupled to and directs the operations of the input former, the composite switch, the prediction processor, and the buffer using computational control signals. The computation resource manager proves particularly effective in systems, such as MCUs, where computation resources are precious.

The computation resource manager **140** continuously monitors the system load by monitoring the input buffer **70** fullness. When the input buffer **70** fullness is high, the system typically cannot process the input data fast enough

and accordingly the load on the system must be reduced. The computation resource manager **140** can modify the load on the system through various means. The computation resource manager **140** can alter the computation resource usage by directing the actions of either the input former **42**, the prediction processor **48**, or the composite switch **50**.

For instance, the computation resource manager **140** can control the composite switch **50** to stop the processing of skipped macroblocks. Additionally, the computation resource manager **140** can instruct the input former **42** to disable the feedback aspects of the system to reduce the computational usage of the transcoder **40**. Particularly, the computation resource manager **140** can force the selector **104** in the input former **42** to form a path between the quantized input signal **124** and the corrected signal **54**. This position of the selector **104** reduces the computational requirements of the system because it cuts off the feedback path generated through the feedback signal **52** and the skipped block signal. Moreover, the computation resource manager **140** can instruct the prediction processor **48** to disable all motion compensation in order to reduce the computational usage of the transcoder **40**.

The computation resource manager **140** can be implemented with a state variable diagram, where the state variable monitors and stores the sum of data indicative of the system load. When the state variable rises above a particular threshold, the computation resource manager **140** acts to reduce the system threshold. The computation resource manager **140** can continue taking further actions as the state variable rises above additional thresholds. In addition, the computation resource manager **140** can remove those aspects of the system which reduce the computational load as the state variable falls. For instance, when the state variable is high, the input former **42** may only generate the corrected signal **54** based upon the quantized input signal **124** and the motion compensator processor may be deactivated. In comparison, when the state variable falls, the computation resource manager **140** may choose to re-activate the prediction processor **48**.

In another aspect of the invention, the computation resource manager **140** can modify the system load by changing the coding mode of the input bitstream **32**. The computation resource manager generates an intra refresh request signal **141** that is coupled with a remote encoder. The remote encoder controls the coding mode of the input bitstream and the intra refresh request signal **141** directs the operation of the remote encoder. Particularly, the control signal **141** can direct the remote encoder to encode the input bitstream **32** in either intra mode or inter mode coding. Accordingly, to reduce the system load the computation resource manager can force the remote encoder to encode the input bitstream in intra mode.

When the computation resource manager forces the transcoder to operate in a computationally less intensive status, the produced picture quality may deteriorate. The deterioration can be stopped by intra coding updating at the remote encoder. However, excessive use of intra-coding can also result in lower picture quality. Hence, the intra mode request signal is only sent by the computation resource manager when the noise level exceeds a predetermined threshold and when the load on the computational usage of the system has exceeded another predetermined level. The noise level can be gauged by providing a counter that counts the number of macroblocks whose coding mode is forced to change during transcoding.

FIG. **13** illustrates a flow chart for implementing the methods steps of the current invention. The method shown transcodes an input signal coded in either intra, inter, or skip block mode. The illustrated method also efficiently transcodes the input signal by selectively providing for motion compensation of the input signal.

The flow chart starts at step **150**. After step **150**, logical flow proceeds to step **152**. At step **152**, the input bitstream is decoded. After decoding, the transcoding system identifies the coding mode of the input bitstream at step **154**. The coding mode can be selected from intra, inter, and skip block modes of coding.

At decision box **156**, the system identifies whether all skipped blocks are completed. If any skipped blocks need to be processed, then logical control proceeds to decision box **158**, otherwise logical control proceeds to decision box **164**.

At decision box **158**, the transcoding system determines whether the current quantinization step size is less than the previous quantinization step size. If the current step size is less, then control proceeds to action box **160**. If the current step size is greater, then control returns to decision box **156**.

At action box **160**, the transcoding system obtains an error macroblock from the buffer **46**. This error macroblock is then applied against the current image signal. After box **160**, control proceeds consecutively to actions box **162**. At box **162**, the transcoding system encodes the output signal **36** using entropy coder **74**. Afterwards, logical flow of the transcoding system returns to decision box **156**.

At decision box **164**, the transcoding system determines whether the coding mode of the input bitstream is intra coding. If the bitstream is not intra coded, i.e. the bitstream is inter coded, then control proceeds to action box **166**. If the bitstream is intra coding, the control proceeds to action box **165** where the coefficients are decoded. After box **165**, logical flow proceeds to action box **168**.

At decision box **174**, the transcoding system determines whether the input bitstream requires motion compensation. If motion compensation is required, then control proceeds to action box **176**, where motion compensation is performed by the prediction processor **48**. If motion compensation is not required, then control proceeds to box **178**. At box **178**, the feedback signal **52** is selectively formed based on either the delayed quantinization error signal **58** or the prediction signal **59**. Logical flow then proceeds to action box **166**.

At box **166**, the input former **42** forms a corrected signal **54** based upon the input bitstream **32** and a feedback signal **52**. Afterwards, at action box **168**, the transcoding system quantizes the corrected signal to form an output signal **36**. After box **168**, logical flow proceeds to action box **170**.

At action box **170**, the system generates an error signal representing the quantinization error of the quantizer used in step **168**. After box **170**, the system buffers the error signal at step **172**. After box **172**, flow proceeds to box **180**.

At box **180**, the entropy coder **74** performs entropy coding on the output signal **36**. At box **182**, the transcoding system proceeds to the next macroblock of data that requires processing.

The invention being thus disclosed and described in connection with the illustrated embodiments, variations and modifications thereof will occur to those skilled in the art, and are intended to be included within the scope of the invention, as defined by the claims appended hereto.

What is claimed is:

1. A transcoder for transcoding an applied digitally encoded input signal and generating an output signal, comprising:

an input former for generating a corrected signal from said input signal and a feedback signal,

a requantizer operably coupled to said input former for quantizing said corrected signal based upon a prede-

termined quantization step size, and requantizer generating said output signal and a quantization error signal,

a buffer operably coupled to said requantizer for receiving said quantization error signal and for generating a delayed quantization error signal,

a prediction processor operably coupled with said buffer for generating a prediction signal as a function of said delayed quantization error signal, and

a composite switch operably coupled with said input former and either of said buffer and said prediction processor, said composite switch forming said feedback signal by selecting between said delayed quantization error signal and said prediction signal,

wherein said prediction processor includes

an inverse transformer operably coupled to receive and perform an inverse linearly transform on said delayed quantization error signal,

a predictor operably coupled with said inverse transformer for receiving a signal from said inverse transformer and generating a spatial prediction signal, and

a transformer operably coupled with said predictor for linearly transforming said spatial prediction signal.

2. A transcoder according to claim **1**, wherein said requantizer further comprises:

a quantizer operably coupled to receive said corrected signal from said input former, said quantizer generating said output signal at a new bit rate,

an inverse quantizer operably coupled with said quantizer for generating a quantized signal, and

an adder operably coupled to receive said quantized signal and said corrected signal, said adder generating said quantinization error signal representative of error introduced by said requantizer.

3. A transcoder according to claim **2**, wherein said quantizer generates a step size signal identifying the current quantinization step size.

4. A transcoder according to claim **1**, wherein said inverse transformer is an inverse discrete cosine transformer and wherein said transformer is a discrete cosine transformer.

5. A transcoder according to claim **1**, wherein said predictor is a motion compensator.

6. A transcoder according to claim **1**, wherein said input former comprises:

an inverse quantizer for forming a quantized input signal, and

a summer for generating an update signal representative of the sum of said feedback signal and said quantized input signal.

7. A transcoder according to claim **6**, wherein said input former further comprises a selector for generating said corrected signal by selecting between said update signal and said quantized input signal.

8. A transcoder according to claim **7**, wherein said input former further comprises a selector for generating said corrected signal by selecting between said update signal, said quantized input signal, and a skip block signal.

9. A transcoder according to claim **1**, wherein said buffer includes a frame buffer for storing said quantinization error signal.

10. A transcoder according to claim **9**, wherein said buffer further comprises

a QP buffer operably coupled for receiving and buffering a step size signal, and

a comparator operably coupled with said QP buffer, and said requantizer said comparator generating a switch control signal based upon a comparison between said step size signal and a buffered step size signal.

11. A transcoder according to claim **1**, wherein said composite switch further comprises

a first switch operably coupled with said buffer and said prediction processor, said first switch forming an intermediate signal by selecting between said delayed quantinization error signal and said prediction signal, and

a second switch operably coupled with said first switch for selectively connecting said intermediate signal between a plurality of output paths.

12. A transcoder according to claim **1**, further comprising a frequency weighted thresholder operably coupled with said input former, said thresholder generating a frequency weighted corrected signal.

13. A transcoder according to claim **12**, wherein said frequency weighted thresholder applies frequency dependent threshold limits to said corrected signal.

14. A transcoder according to claim **1**, further comprising a resource manager for monitoring the computational usage status of said transcoder and for controlling said transcoder in response thereto.

15. A transcoder according to claim **14**, wherein said computation resource manager monitors an input buffer fullness.

16. A transcoder according to claim **14**, wherein said computation resource manager controls said prediction processor to modify the computational usage of said transcoder.

17. A transcoder according to claim **14**, wherein said computation resource manager controls said prediction processor to modify the computational usage of said transcoder.

18. A transcoder according to claim **14**, wherein said computation resource manager controls said composite switch to modify the computational usage of said transcoder.

19. A transcoder according to claim **14**, wherein said computation resource manager is operably coupled with a remote encoder and wherein said computation resource manager controls the coding mode of said remote encoder.

20. A transcoder according to claim **19** further comprising a counter, wherein said computation resource manager generates an intra refresh request signal for controlling said remote encoder when said counter exceeds a predetermined threshold.

21. A transcoder according to claim **20**, wherein said counter counts the number of macroblocks affected by computational overload.

22. A method for transcoding an applied digitally encoded input signal and generating an output signal, said method comprising the successive steps of:

generating a corrected signal as a function of said input signal and a feedback signal,

quantizing said corrected signal based upon a predetermined quantization step size to from said output signal,

determining a quantization error signal based upon said output signal and said corrected signal,

buffering said quantization error signal and generating a delayed quantization error signal,

forming a prediction signal as a function of said delayed quantization error signal, and

determining a feedback signal by selecting between said delayed quantization error signal and said prediction signal,

wherein said step of forming a prediction signal includes generating a spatial signal by inverse linearly transforming said delayed quantization error signal,

generating a spatial prediction signal by compensating said transformed signal with a motion vector, and

generating said prediction signal by linearly transforming said spatially corrected signal.

23. A method according to claim 22, further comprising the step of identifying the coding mode of said input signal.

24. A method according to claim 23, wherein said step of generating said corrected signal further comprises selectively forming said corrected signal depending upon said identified coding mode.

25. A method according to claim 24, further comprising the steps of:

generating an update signal by summing said feedback signal and said input signal, and

forming said corrected signal as a function of said update signal when said identified coding mode is inter mode.

26. A method according to claim 24, further comprising the step of forming said corrected signal as a function of a quantized input signal when said identified coding mode is intra mode.

27. A method according to claim 26, further comprising the steps of:

forming a skip block signal, and

determining said corrected signal as a function of a skip block signal when said identified coding mode is skipped block mode.

28. A method according to claim 27, wherein said step of generating said skip block signal further comprises the steps of:

generating a step size signal identifying the current quantization step size,

forming a switch control signal as a function of said generated step size signal, said switch control signal directing the formation of said skip block signal.

29. A method according to claim 22, wherein said step of quantizing said corrected signal further includes weighting said corrected signal with frequency dependent threshold limits.

30. A method according to claim 22, further comprising the steps of monitoring and controlling a computational usage status.

31. A method according to claim 30, further comprising the steps of:

incrementing a state variable as said computational usage status rises above a predetermined level, and

reducing said state variable as said computational usage status falls below said predetermined level.

32. A method according to claim 31, wherein said step of controlling said computational usage status further comprises selecting a signal path for generating said corrected signal as a function of said state variable.

33. A method according to claim 31, wherein said step of controlling said computational usage status further comprises selecting a signal path for generating said feedback signal as a function of said state variable.

34. A method according to claim 31, wherein said step of controlling said computational usage status further comprises generating a computational control signal as a function of said state variable, said computational control signal controlling at least one element selected from the group consisting of an input former, a composite switch, a buffer, and a prediction processor.

35. A method according to claim 30, wherein said controlling step further includes controlling the coding mode of a remote encoder.

36. A method according to claim 35, further including the step of generating an intra refresh request signal for controlling said remote encoder when said counter exceeds a predetermined threshold.

37. A method according to claim 36, further including the step of counting the number of macroblocks affected by computational overload.

* * * * *

US005121191A

United States Patent [19]

Cassereau et al.

[11] Patent Number: 5,121,191

[45] Date of Patent: Jun. 9, 1992

[54] METHOD AND APPARATUS FOR CODING MOTION PICTURES

[75] Inventors: Philippe M. Cassereau, Nashua, N.H.; Johnathan D. Devine, Boston, Mass.; John C. Huffman, Windham, N.H.; William R. Zettler, Jr., Brookline, Mass.

[73] Assignee: Aware, Inc., Combridge, Mass.

[21] Appl. No.: 669,773

[22] Filed: Mar. 15, 1991

[51] Int. Cl.[5] H04N 11/04; H04N 9/80; H04N 7/13

[52] U.S. Cl. 358/13; 358/133; 358/136; 358/339

[58] Field of Search 358/11, 12, 13, 105, 358/133, 135, 136, 339

[56] References Cited

U.S. PATENT DOCUMENTS

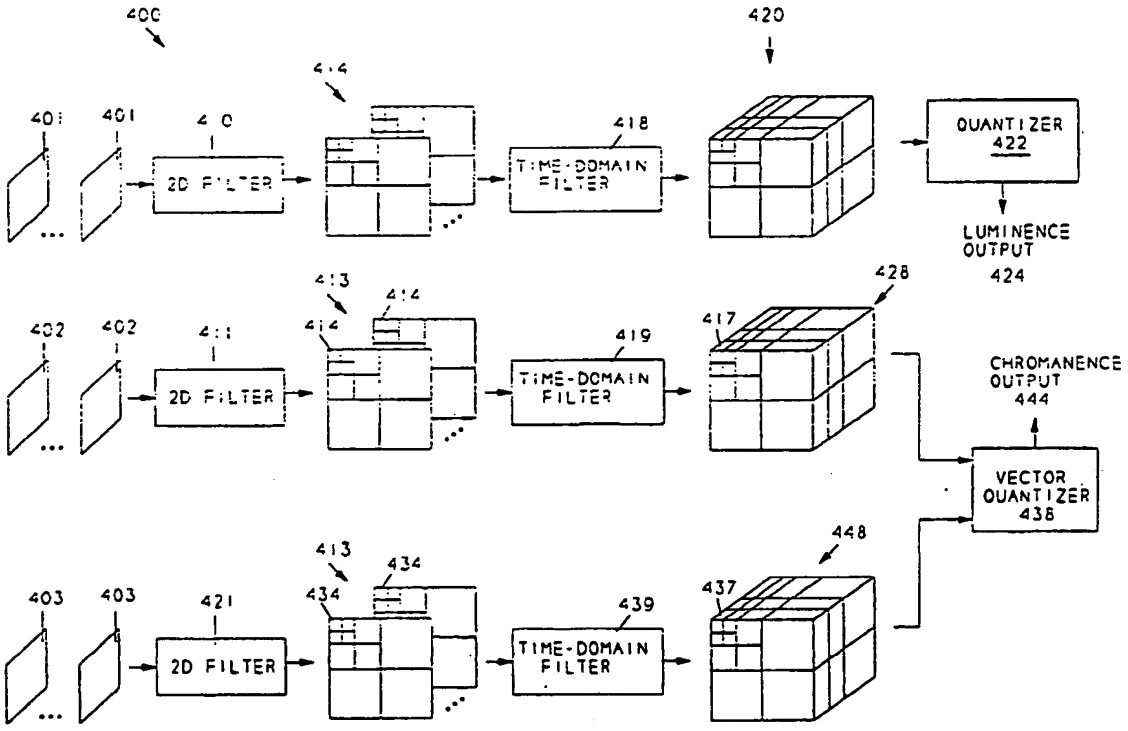
4,663,660 5/1987 Fedele 358/13
4,785,349 11/1988 Keith 358/13
4,827,336 5/1989 Acampora 358/13
4,864,398 5/1989 Avis 358/136

Primary Examiner—Howard W. Britton
Attorney, Agent, or Firm—McCubbrey, Bartles, Meyer & Ward

[57] ABSTRACT

A method and apparatus for compressing motion pictures is disclosed. The frames of the motion picture are divided into adjacent groups. Each group is treated as a three-dimensional image. The three-dimensional image is then filtered via a three-dimensional FIR filter to generate three-dimensional component images that are more efficiently quantized. The degree of quantization of each component image is determined in part by the spatial frequencies represented by the component image in question. For motion pictures derrived from interlaced scanning devices, the quantization of specific component images is altered to prevent artifacts.

17 Claims, 8 Drawing Sheets



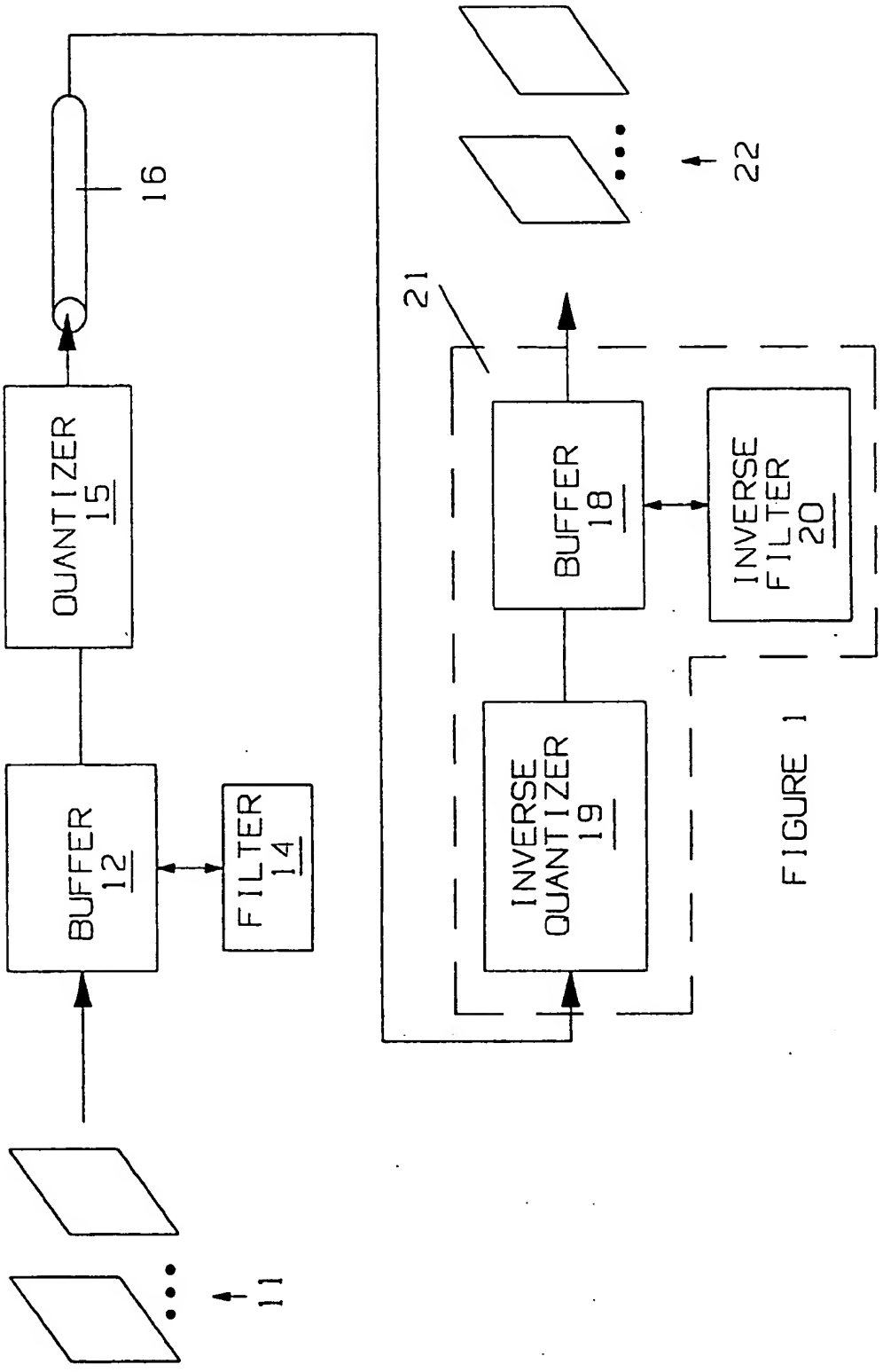

11
BUFFER 12
FILTER 14
QUANTIZER 15
16
22
21
BUFFER 18
INVERSE FILTER 20
INVERSE QUANTIZER 19

FIGURE 1

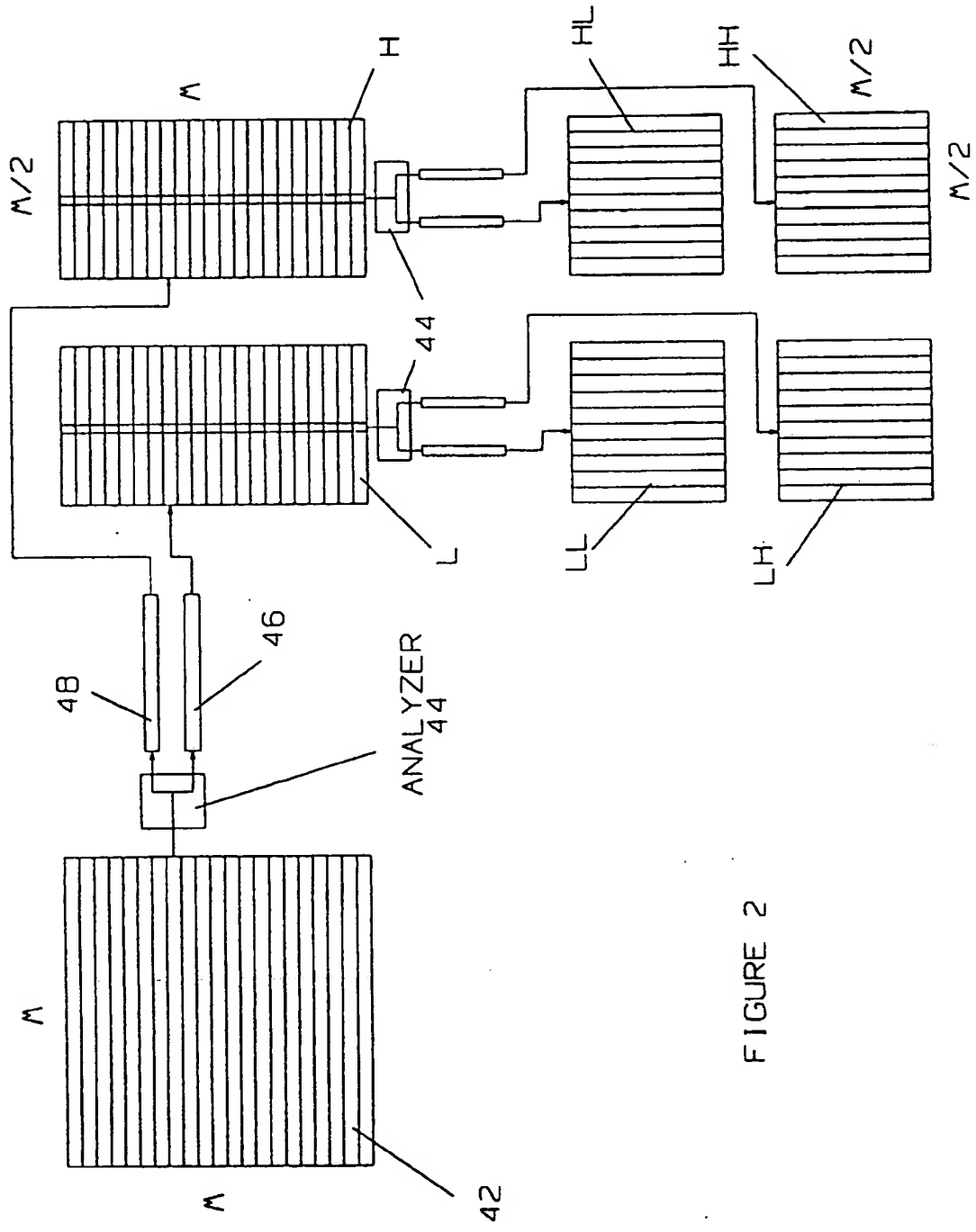


FIGURE 2

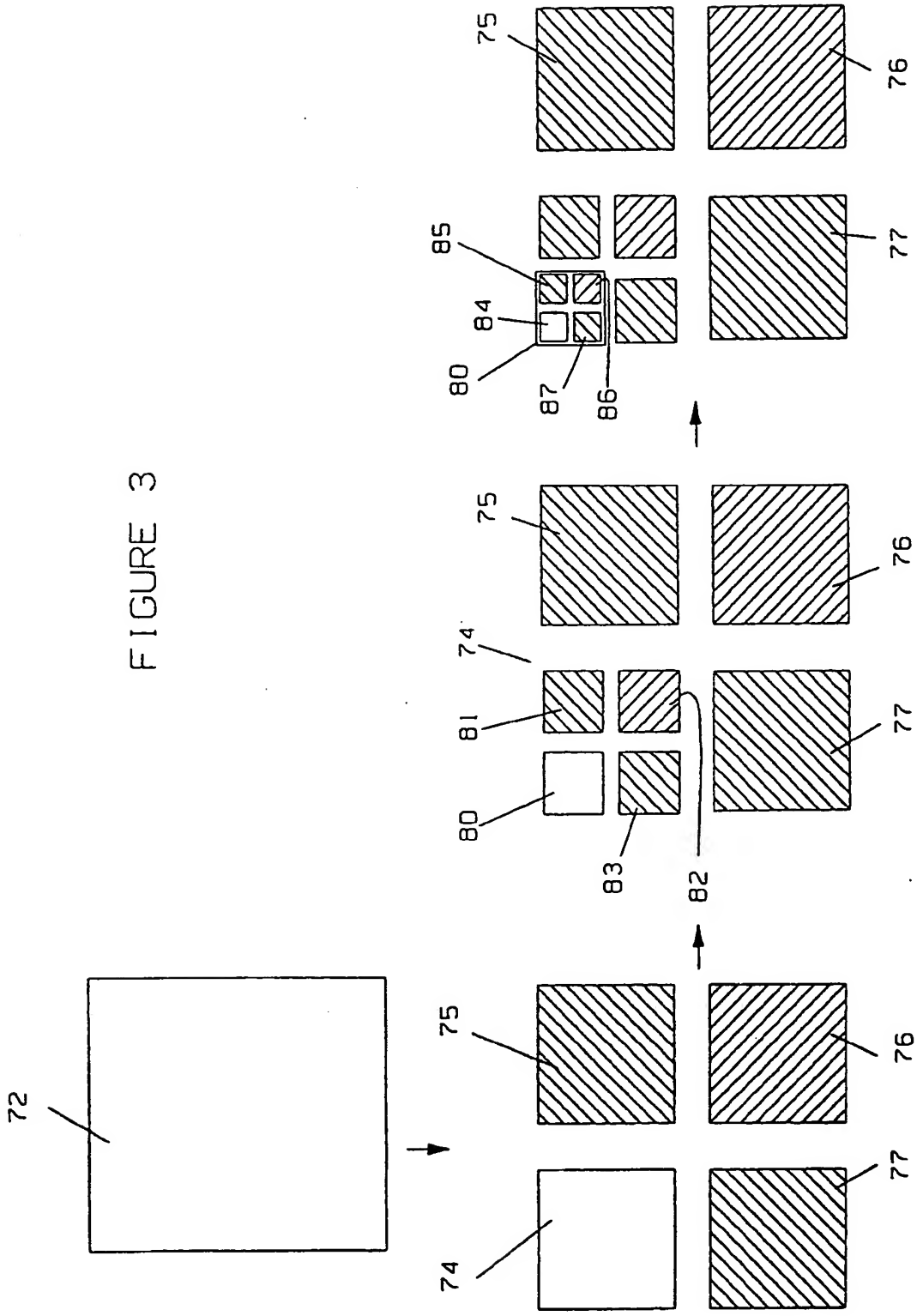


FIGURE 3

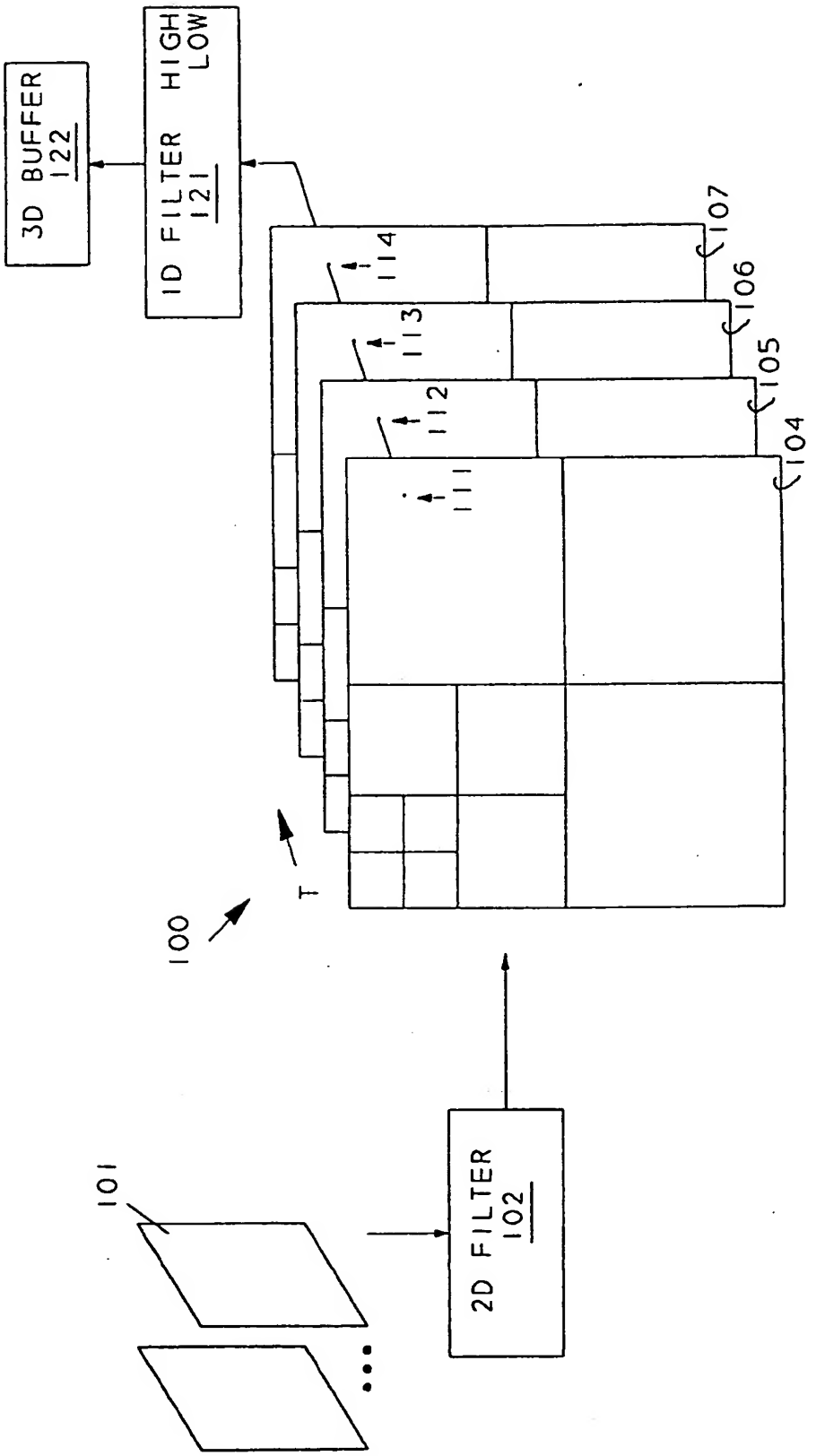

3D BUFFER
122
1D FILTER HIGH LOW
121
114
113
112
111
107
106
105
104
T
100
101
2D FILTER
102

FIGURE 4

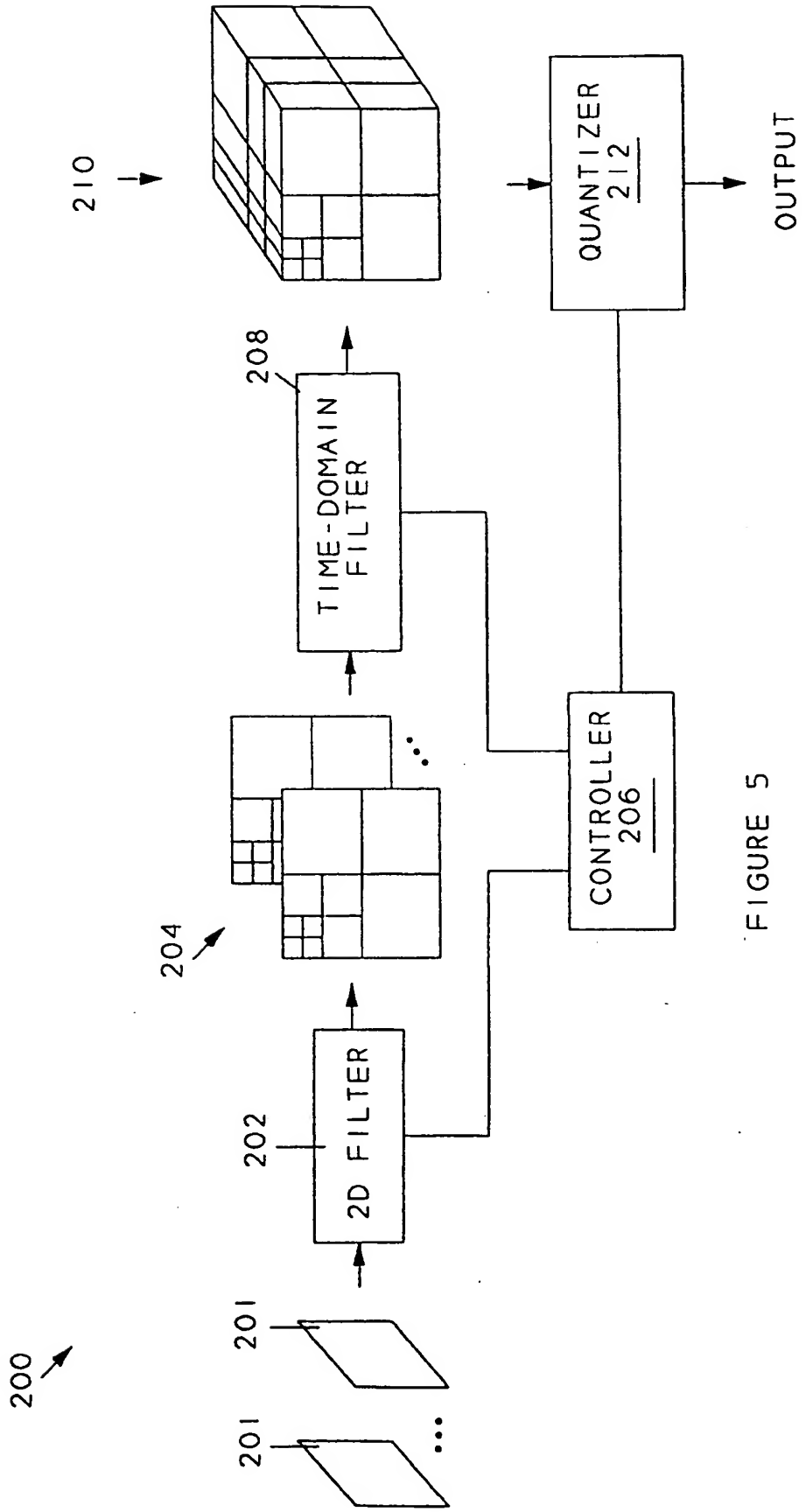

200
201
201
202
2D FILTER
204
208
TIME-DOMAIN FILTER
210
CONTROLLER 206
QUANTIZER 212
OUTPUT

FIGURE 5

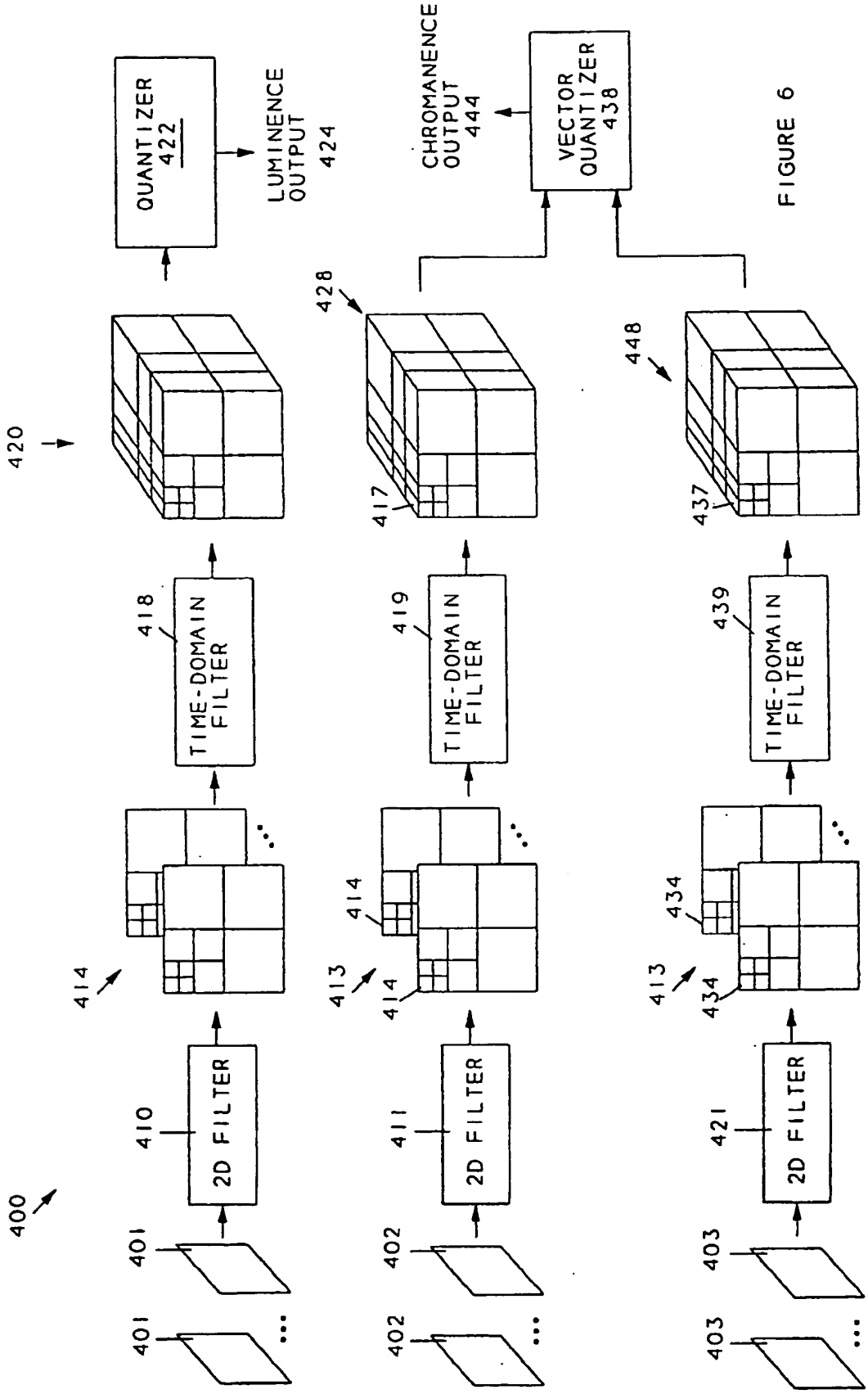

400
401
401
2D FILTER
410
TIME-DOMAIN FILTER
418
414
420
QUANTIZER 422
LUMINENCE OUTPUT 424
402
402
2D FILTER
411
TIME-DOMAIN FILTER
419
413
414
414
417
428
CHROMANENCE OUTPUT 444
VECTOR QUANTIZER 438
403
403
2D FILTER
421
TIME-DOMAIN FILTER
439
413
434
434
437
448

FIGURE 6

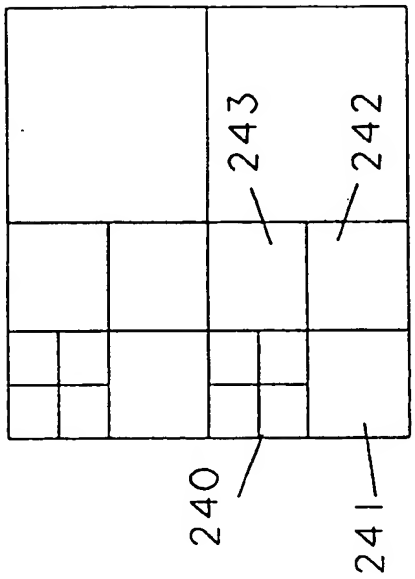


FIGURE 8

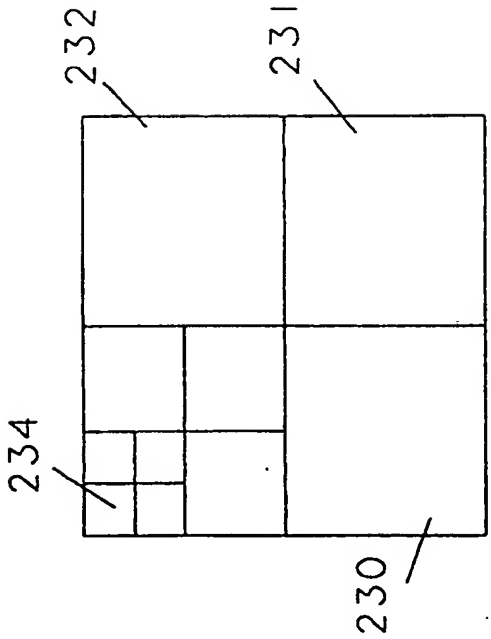


FIGURE 7

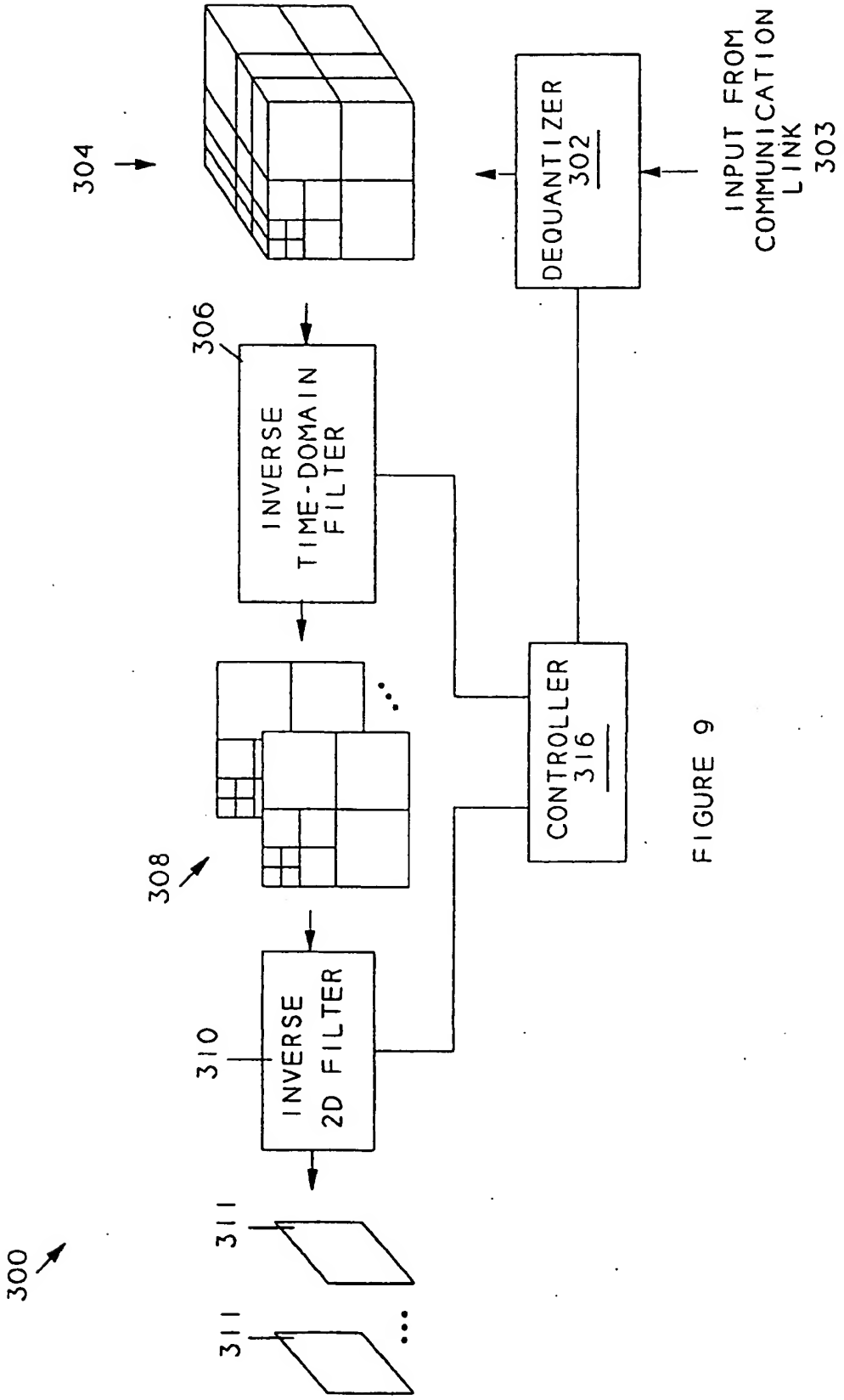

304
DEQUANTIZER 302
INPUT FROM COMMUNICATION LINK 303
306
INVERSE TIME-DOMAIN FILTER
CONTROLLER 316
308
310
INVERSE 2D FILTER
311
311
300

FIGURE 9

METHOD AND APPARATUS FOR CODING MOTION PICTURES

BACKGROUND OF THE INVENTION

The present invention relates to methods and apparatuses for reducing the amount of data needed to represent an image, and more particularly, to a novel method and apparatus for compressing motion pictures.

Images are conventionally represented by a two-dimensional array of values in which each value represents a property of the image at a corresponding point on the image. In the case of gray-scale images, a single number representing the gradations of intensity from white to black, referred to as the gray scale, is stored. In the case of color images, each "value" is a vector whose components represent the gradations in intensity of the various primary colors, or some alternative color code, at the corresponding point in the image.

A motion picture comprises a sequence of such images. Typically, thirty or more images are needed for each second of viewing time.

This representation of a motion picture corresponds to the output of a typical image-sensing device such as a television camera. Such a representation is convenient in that it is easily regenerated on a display device such as a CRT tube. However, the number of bits needed to represent the data is prohibitively large for many applications. A single 512×512 gray-scale image with 256 gray levels requires in excess of 256,000 bytes. At 30 frames per second, a communication channel with a bandwidth of approximately 64 million bits per second is needed to transmit the motion picture. A full color 24 bit per pixel motion picture would require a bandwidth of more than 190 million bits per second.

This bandwidth is significantly greater than that available for many communications purposes. For example, high quality consumer telephone lines are typically limited to 64 thousand bits per second. Hence, commercially viable picture telephone systems require some form of image compression.

Image compression systems used in motion picture compression applications make use of the redundancy between frames to reduce the amount of information that must be transmitted over the bandwidth-limited communication link. Many scenes in a motion picture include portions that are constant from frame to frame for several seconds, if not minutes. Prior art compression schemes attempt to divide the scene into the constant portion and the portion that changes from frame to frame.

The constant portion need be sent only once. Hence, if the constant portion comprises a significant fraction of the scene, considerable image compression can be realized. For example, if the entire scene were constant for 10 seconds, the information needed to reproduce the scene would be contained in one frame of the sequence and an indication of the number of times the frame is to be repeated. This would be approximately 1/300 of the information needed if the scene were sent without compression.

To better understand the manner in which this type of prior art image compression operates, consider a motion picture which is being sent over a bandwidth-limited communication link from a transmitter to a receiver. The receiver will be assumed to have a frame buffer which stores one frame of the motion picture. The contents of this frame are then displayed at the receiver. Ideally, the frame in the receiver will exactly match the corresponding frame at the transmitter.

Typically, the process is started by sending one complete frame of the sequence. This frame is referred to as the key frame. After the key frame is sent, successive frames are specified by sending information specifying changes that have occurred relative to the last frame displayed by the receiver. The receiver uses this information to update its frame buffer. If the interframe changes are relatively small, i.e., the scene is almost constant, the available bandwidth will be sufficient to transmit all of the information needed to update the receiver's image. That is, the frame buffer in the receiver will exactly match the corresponding frame of the motion picture at the transmitter.

Unfortunately, the available bandwidth is often insufficient to completely specify all of the interframe changes that have occurred since the last update in 1/30 of a second. Hence, the changes to be sent are prioritized and only the highest priority changes are sent. As a result, errors accumulate in the frame buffer at the receiver. The transmitter keeps track of the magnitude of these errors.

When these errors become sufficiently large, a new key frame must be sent. The time needed to send a key frame is greater than 1/30 of a second. Hence, each time a key frame is sent, the motion will be interrupted for some period of time. For example, at a resolution of 512×512 pixels each requiring 8 bits, a key frame would require between one and two seconds to send at 1.44 Mbps, a data rate typically used in teleconference systems. During this time interval additional frames will have been generated at the transmitter. The corrections needed to bring the receiver frame buffer up to date must be sent as a single large update. The combination of the time delay and the large update results in a noticeable loss of continuity in the motion.

The various errors and motion interruptions become particularly objectionable when the sequence of scenes in the motion picture change rapidly. To minimize these problems, either the bandwidth of the communication channel must be increased or the quality of the picture must be reduced. Providing increased bandwidth increases the cost of transmitting the motion picture, and hence, is to be avoided. Similarly, a reduction in picture quality is also to be avoided.

A second problem with this type of prior art motion picture compression system occurs in systems using interlaced video. In interlaced video systems, successive frames are offset relative to one another by an amount to one half the horizontal line spacing in a frame. Hence, the frames do not "line up" precisely when considered as sequence. Prior art systems do not provide an acceptable method for compensating for this motion. As a result, the decompressed motion picture often appears to have jitter.

Broadly, it is an object of the present invention to provide an improved apparatus and method for compressing motion pictures.

It is a further object of the present invention to provide a method and apparatus which is less sensitive to scene changes than prior art systems.

It is a still further object of the present invention to provide a method and apparatus which is more immune to compression artifacts in interlaced video systems than prior art systems.

These and other objects of the present invention will become apparent to those skilled in the art from the following detailed description of the invention and the accompanying drawings.

SUMMARY OF THE INVENTION

The present invention includes both apparatuses and methods for compressing a motion picture. An apparatus for compressing a motion picture according to the present invention operates on a motion picture comprising a sequence of frames, each said frame comprising a two-dimensional image representing said motion picture at one instant in time. The apparatus includes circuitry for receiving a signal specifying a compression ratio for said motion picture and for receiving signals specifying said frames. A predetermined number N_f of consecutively received frames are stored by the apparatus. These frames are processed by a three-dimensional filter which generates therefrom a plurality of three-dimensional component images. The plurality of three-dimensional component images includes a low-frequency component image and one or more high-frequency component images. Each of the component image comprises a plurality of pixel values. The compression apparatus also includes quantization circuitry for approximating each of the pixel values in at least one of said component images by an approximation having one of a predetermined set of integer values. The precision of said integers depends on the statistical distribution of the pixel values in said component image and said received compression ratio. These pixel approximations specify an approximation to the original N_f frames of the motion picture. In the preferred embodiment of the present invention, the pixels of the low-frequency component image are approximated to a precision not less than the statistically optimum precision determined by said statistical distributions of said pixels in the component images and the received compression ratio and the pixels of at least one of said high-frequency component images are approximated to a precision not greater than the statistically optimum precision determined by the statistical distribution of the pixels in the component images and the received compression ratio.

In the embodiments of the present invention in which color motion pictures are compressed, additional compression is obtained by compressing the chrominance signals to a greater degree than the luminance signals. In addition, the pixels of the chrominance signals are quantized and/or coded using vector techniques.

By selectively retaining certain high-frequency information, significantly greater compression is obtained for motion pictures in an interlaced format.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a video transmission utilizing the compression technique of the present invention.

FIG. 2 is a block diagram of an apparatus for filtering each frame of a motion picture to generate a plurality of two-dimensional component images.

FIG. 3 illustrates the manner in which the apparatus shown in FIG. 2 may be iteratively applied to generate additional two-dimensional component images of having different frequency content.

FIG. 4 illustrates the manner in which a two-dimensional filter and a one-dimensional filter are combined so as to generate the three-dimensional filtered component images utilized by the present invention.

FIG. 5 is a block image compression apparatus according to the present invention.

FIG. 6 illustrates the manner in which a color motion picture is compressed by an apparatus according to the present invention.

FIG. 7 illustrates the manner in which a single frame of a non-interlaced motion picture is filtered.

FIG. 8 illustrates the manner in which a single frame of an interlaced motion picture is filtered.

FIG. 9 illustrates the manner in which a motion picture that has been compressed by the method of the present invention is decompressed.

DETAILED DESCRIPTION OF THE INVENTION

To simplify the following discussion, it will be assumed that a monochrome motion picture is to be compressed. The manner in which information specifying color is compressed will be discussed afterward. The compression technique of the present invention may be more easily understood with reference to FIG. 1 which is a block diagram of a video transmission apparatus utilizing the compression technique of the present invention. The motion picture consists of a sequence of two-dimensional images shown at 11. Each image comprises a two-dimensional array of intensity values $I_{x,y}$, where the x and y values correspond to points on a two-dimensional grid. Each array represents the motion picture at some time t. Hence, the motion picture may be represented by a three-dimensional array $I_{x,y,t}$. At any given time, compression apparatus 13 stores and processes a predetermined number of consecutive frames of the motion picture. The stored frames are held in a buffer 12. The stored frames are considered to be a single three-dimensional image.

A three-dimensional perfect reconstruction filter 14 operates on the stored three-dimensional image. The filter may be viewed as creating a number of component images. Each component image has information corresponding to information in specific scales, which correspond approximately to specific spatial frequency bands in the original three-dimensional image. The filtered images may be recombined using an inverse filter to regenerate the original three-dimensional image from the component images provided the filter and its inverse have sufficient numerical accuracy.

It should be noted that the filtering transformation described above does not in itself provide image compression. If the filtered component images are represent to the same precision as the original frames of the motion picture, the number of bits needed to store the filtered component images is the same as that needed to store the original motion picture frames. The actual image compression results from coding operations performed on the different filtered component images.

The process by which the intensity values in the filtered images are coded will be referred to as quantization. Quantizer 15 replaces each suitably normalized intensity value by an integer which approximates the normalized intensity value. The error introduced by this approximation will depend on the range of intensity values and the number of bits allocated to each integer. If the range of intensity values is large, the number of bits needed to adequately approximate the intensity values will be larger than would be the case if the range of intensity values were small.

Consider the pixel values of one of the component images. Each of the pixel values is to be replaced by an

integer having a predetermined number P of bits. The integers in question are then transmitted in place of the individual pixel values. At the receiver, the inverse of the mapping used to assign the integer values to the pixel values is used to recover the original pixel values. There are at most 2^P distinct values that can be represented; hence, if there are more than 2^P different pixel values in the sub-set of pixel values, at least some of the pixel values will not be exactly recovered. The goal of the quantization algorithm is to minimize the overall error rate with respect to a suitable measure of error. Given the statistical distribution of the pixel values and the average number of bits to be allocated to each pixel, the optimum mapping of the pixel values to the integer values in question may be calculated. Since such optimization calculations are well known in the art, they will not be discussed in detail here. For more details of the procedures in question, the reader is referred to William K. Pratt, *Digital Image Processing*, pp. 141–46, John Wiley & Sons, Inc., New York, 1978, which is hereby incorporated by reference.

The overall error in approximating a component image will depend on the statistical distribution of the intensity values in the component images and the degree of image compression needed. The ratio of the number of bits needed to represent the original three-dimensional image to the number of bits in the compressed image will be referred to as the image compression ratio. Once an image compression ratio is set, the total number of bits available for all of the component images is determined. The optimum assignment of the available bits to the various component images is then determined from the statistical properties of the component images. It has been found experimentally that the statistical distributions of the component images generated by the present invention are approximated by Laplacian distributions. Hence, the optimum allocation of the bits may be made from a knowledge of the variance of the pixel values in each component image. Schemes for optimally allocating bits are known to those skilled in the art, and hence, the procedures for converting the observed variances to quantization levels will not be described in more detail. For the purposes of this discussion, a quantization scheme will be defined to be "optimum" if it provides the lowest distortion in the reconstructed image for a given compression ratio. For the purposes of this definition, the distortion is measured in terms of mean-squared-error between the frames of the original motion picture and those of the motion picture obtained after compressing and decompressing the original motion picture. It will be apparent to those skilled in the art that other statistical measures of distortion between the original and reconstructed motion pictures may be used.

In the preferred embodiment of the present invention, the various component images are quantized using a uniform quantization scheme. In such schemes, the intensity of each pixel is transformed into an integer which is equal to the intensity value in question divided by a step size. In general, the step size will vary between component images. In the preferred embodiment of the present invention, the step sizes used for the various component images are determined by assuming a Laplacian distribution for the pixel values. The optimum step sizes are then determined from the variances of the pixels values of the component images, the bits needed to represent the original image, and the desired compression ratio.

The filtering of the three-dimensional image into component images of different spatial and time domain frequencies serves two functions in the present invention. First, the filtering operation reduces the correlation of the pixels in space and time. This reduction in correlation leads to reduced variances, and hence, reduces the number of bits needed to adequately represent the various component images. Most motion picture images are highly correlated both in space and time. Hence, the three-dimensional images generated from a number of consecutive frames will be highly correlated in all three dimensions. If the three-dimensional image can be transformed utilizing a linear transformation in space and time such that the transformed image has reduced correlation, then the variance of the pixels of the transformed image will be less than that of the original image. In this representation, relatively few of the pixels in the transformed image will contain most of the image signal energy, and the remaining pixels will have reduced variance and, hence, can be coded with fewer bits. The filtering operation utilized in the present invention provides this transformation.

It is found experimentally that the lowest frequency component image has a variance at least as large as the original three-dimensional image; however, the various high-frequency images have significantly lower variances. As will be explained in more detail below, the total number of pixels in the various component images is approximately the same as that in the original image. However, the number of pixels in the lowest frequency component image is a small fraction of the total pixels in the transformed image. Hence, even if pixels of the lowest frequency component image are quantized to an accuracy equal to or greater than that of the original image, the more numerous pixels of the various high-frequency component images can be quantized to an accuracy much less than that of the original image. As a result, a net image compression will be obtained.

Second, the filtering operation separates the image information into subsets which have different subjective value to a human observer. The three-dimensional images processed by the present invention have two spatial dimensions and one time dimension. It is known from the study of human vision that low spatial frequency information is more important to a viewer than high spatial frequency information when viewing still photographs. It is also known from the study of vision that the human eye cannot resolve fine detail in moving objects. Hence, objects in an image that are moving can be displayed with lower precision than stationary objects. Thus, the time dimension behaves in the same manner as the spatial dimensions of the individual frames in that low frequency information is more important than high frequency information. In the preferred embodiment of the present invention, more bits are allocated to the low frequency component image than would be allocated using a statistically optimum quantization scheme. This allocation preferentially retains the information most important to a human observer. In addition, fewer bits per pixel are allocated, on average, to the high frequency component images than would be allocated by a statistically optimum quantization scheme. This allocation improves the image compression by selectively degrading the information which is less important to a human observer. In the preferred embodiment of the present invention, some of the highest frequency component images are not quantized at all. The non-quantized component images are

discarded. Since the number of pixels in the low frequency component image is a small fraction of the number of pixels in the various high frequency component images, additional compression is obtained for a given degree of subjective image quality.

Referring again to FIG. 1, the component images, after quantization, are transmitted over a bandwidth limited communication channel 16. At the receiving end of the channel, the inverse three-dimensional filter is applied to reconstruct an approximation to the original motion picture by a decompression apparatus 21. The bandwidth limited channel may include storage devices such as magnetic disks or other devices of finite capacity. Hence, the method of the present invention may also be used to store motion pictures in a form requiring less storage medium than would be needed in the absence of compression.

Decompression apparatus 21 performs the inverse of the transformations carried out by compression apparatus 13. The quantized component images are received by an inverse quantizer 19 which generates intensity levels from the quantized values. Inverse quantizer 19 inserts zeros for the pixels of the component images that were not quantized. The resulting three-dimensional component images will be approximations to the original three-dimensional component images produced by filter 14. The intensity values in question are stored in a buffer 18. An inverse perfect reconstruction filter 20 then constructs a single three-dimensional image from the component images. The single three-dimensional image in question is then divided into a plurality of frames which are then outputted as shown at 22.

In general, the three-dimensional filter operates on a three-dimensional image to generate a plurality of three-dimensional component images which may be recombined by the inverse filter to regenerate the original image. The preferred filter creates a low-frequency component image and a plurality of high frequency component images. The low-frequency component image will have an average pixel intensity value that is greater than zero, whereas the average pixel intensity value in the high frequency images will be zero. The high-frequency component images preferably have lower correlation than the original image. In addition, the frequency content of the various high frequency images is preferably different for different sets of high-frequency component images. That is, the collection of high-frequency component images may be organized into groups having component images with different frequency content. Within a group, different component images may emphasize different features of the images. For example, one component image may contain information relating to horizontal lines in the image, while a second component image may have the same frequency content but contain information relating to vertical edges. That is, the high frequency component image emphasizing horizontal edges would contain a line segment at the location of a horizontal edge if the filter preserves local structure.

Finally, it is advantageous to choose filters that are sensitive to the types of spatial and time variations that are normally encountered in motion pictures. It may be shown that the filtering operation described above is equivalent to approximating the original image by a linear superposition of basis functions. That is, an image represented by an array $I_{x,y,t}$ may be approximated by a series of the form

$$I_{x,y,t} = \sum_i \sum_j \sum_k \sum_m {}^{m}C_{i,j,k}\,{}^{m}F_{i,j,k}(x,y,t)$$

The set of functions $\{{}^{m}F_{i,j,k}(x,y,t)\}$ forms a basis for a vector space. The various values of m correspond to the different component images, and the coefficients ${}^{m}C_{i,j,k}$ are the pixels of the various component images.

For image compression applications, the basis functions preferably have a number of additional properties. First, it is advantageous to utilize an approximation in which the basis functions form an orthonormal basis. It can be shown that an orthonormal basis requires fewer coefficients ${}^{m}C_{i,j,k}$ to adequately represent the image than a non-orthonormal basis having similar spatial properties. Hence, a filter based on an orthonormal basis concentrates the information in component images having fewer pixels which, in turn, permits a higher compression ratio for any given image quality.

Second, it is advantageous to use a basis in which the basis functions have properties which permit efficient representation of image information found in the motion picture. If this property is satisfied, fewer coefficients ${}^{m}C_{i,j,k}$ will be needed to adequately represent the image than would be the case if basis functions having properties that are radically different from those of the objects in the motion picture are used. For example, most images, and hence most motion pictures, are made up of objects whose intensity varies smoothly over the object. Hence, a basis in which the basis functions approximate polynomials in the spatial dimensions is desirable. In addition, most motion pictures have edges in the spatial dimensions and include scenes that are relatively constant in time for some period and then suddenly change. Hence, it is desirable to utilize a basis which can adequately approximate edges in space and time. It has been found experimentally that these properties can be adequately satisfied by using basis functions that approximate low order polynomials. In particular, basis functions approximating quadratics or cubics in the spatial dimensions are found to be optimum.

Third, the spatial extent of each basis function should be limited. The portion of the image over which a given basis function is non-zero is referred to as the support of the basis function. If the support of each of the basis functions ${}^{m}F_{i,j,k}(x,y,t)$ is small compared to the image being approximated, an error in the corresponding coefficient ${}^{m}C_{i,j,k}$ introduced by the quantization process will only affect a small fraction of the reconstructed image. In this case, random errors in the coefficients will result in noise-like errors in the reconstructed image. If, however, the support of the basis functions is large, a quantization error in a coefficient will lead to an error in the reconstructed image that affects a large fraction of the image. Such errors may appear as stripes or other artifacts. It is known from physiological research that noise-like errors are far less objectionable to a human observer than coherent artifacts such as stripes. Hence, it is advantageous to utilize a filter that corresponds to a functional expansion employing basis functions having limited support.

The manner in which the perfect reconstruction filter and its inverse are constructed will now be explained in more detail. In the preferred embodiment of the present invention, three-dimensional filter 14 is constructed by iteratively applying a one-dimensional perfect reconstruction filter to the three-dimensional image. The

preferred filter is a conjugate mirror filter bank that generates two outputs from an input signal comprising a linear sequence of signal values. The outputs in question comprise low-frequency and high-frequency components of the input signal. In the preferred embodiment of the present invention, the filter bank is implemented as digital finite impulse response filters (FIR). Hence, the conjugate mirror filter bank accepts a vector having M components, x(i). If M is not even, a zero is appended to x to force M to be even. The filter bank multiplies the elements of x with the elements of a vector having components a(k), where k runs from 0 to N−1, to form two vectors having components, $y_L(i)$ and $y_H(i)$. These vectors each have M/2 such elements. The components of $y_L(i)$ and $y_H(i)$ are calculated as follows:

$$y_L(2m) = \sum_{k=0}^{N-1} a(k)x(2m - N - k) \quad (1)$$

$$y_H(2m + 1) = \sum_{k=0}^{N-1} a(k)(-1)^{k-1} x(2m + 1 - k) \quad (2)$$

Here, m runs from 0 to (M/2)−1.

The transformations shown in Eqs. (1) and (2) require that some value be assigned to pixels that lie outside the signal range, i.e., x(i) for i<0 or i>M−1. In the preferred embodiment of the present invention, x is assumed to be cyclic, i.e., x(i)=x(i+M). In this case, it can be shown that y_L and y_H each have at most M/2 non-zero components, and the total number of pixels in the various component images will be the same as that in the original image. Other possible assumptions for the values outside the defined range of x(i) will be apparent to those skilled in the art. For example, the values outside the range could be set to zero. In this case, the maximum number of non-zero components will be approximately M/2.

The coefficients a(k) shown in Eqs. (1) and (2) are real or complex numbers that must satisfy the following relationship for the filter bank to be a conjugate mirror filter bank:

$$\sum_{i=0}^{N-1} a^*(i)a(i + k) = 2\delta_{k,0} \quad (3)$$

Here, a(k)=0 for k<0 or k>(N−1), and k is any even integer, and a*(i) denotes the complex conjugate of the (possibly) complex number a(i). It may be shown that filters satisfying Eq. (3) are equivalent to expanding the input signal x(i) in a orthonormal set of basis functions. It can be shown that the three-dimensional filter obtained by the iterative application of such a one-dimensional filter is equivalent to expanding the three-dimensional image in a three-dimensional orthonormal basis of the type discussed above. In addition, for the filter to generate high and low frequency components, the following relationship must be satisfied.

$$\sum_{k=0}^{N-1} a(k) = 2 \quad (4)$$

It may be shown that the support of the basis functions is determined by the number of a(i) which are non-zero. Hence, filters having a small number of weights are preferred. Filters in which the weights satisfy Eqs. (3) and (4) will provide good polynomial approximations if they also satisfy the following system of equations

$$\sum_{k=0}^{N-1} (-1)^k k^m a(k) = 0 \quad (4a)$$

for m=0,1,..., D where D is the degree of the desired polynomial approximation. An exemplary set of filter coefficients having N=6 is as follows:

k	a(k)
0	.3327
1	.8069
2	.4599
3	−.1350
4	−.0854
5	.0352

In the preferred embodiment of the present invention, each frame of the motion picture to be included in a particular three-dimensional image is first filtered to form a plurality of two-dimensional filtered component images. Corresponding two-dimensional filtered component images in a plurality of these frames are then filtered to form the plurality of three-dimensional filtered component images discussed above.

The manner in which each frame is coded as a plurality of two-dimensional filtered component images will now be explained with reference to FIGS. 2 and 3. Each frame of the motion picture is assumed to be represented by an array of pixel values having M columns and M rows. The generalization of the method taught below to non-square arrays of pixels will be apparent to those skilled in the art from the following discussion. Each frame is coded by iteration of a two-dimensional filtering transformation. The two-dimensional transformation is carried out by applying a one-dimensional transformation twice as indicated in FIG. 2. In the first step, an image 42 comprising a single frame of the motion picture with M columns and M rows of pixels is transformed into two images denoted by L and H by a filter circuit 44 which will be referred to as analyzer 44. Analyzer 44 comprises a conjugate mirror filter bank as described above. The L and H images each have (M/2) columns and M rows. Each row in the L and H images is generated by filtering the corresponding row in the image 42 through analyzer 44. The high frequency output of analyzer 44 becomes the corresponding row in the H image, and the low frequency output of analyzer 44 becomes the corresponding row in the L image.

In the second step in the transformation, each of the L and H images is further transformed into two (M/2)×(M/2) component images. Image L is used to create two images, LL and LH in a manner analogous to that described above, except that the image is divided into columns instead of rows. Each image is divided into its M element column vectors. Each vector is transformed using Analyzer 44 into two M/2 element column vectors having elements y_L and y_H using the transformation defined in Eq. (1). The vector y_L obtained by transforming the i^{th} column of image L becomes the i^{th} column of image LL. Similarly, the vector y_H becomes the i^{th} column of image LH. The remaining images HL and HH are obtained in an analogous manner.

Upon completion of the transformation, the original frame will have been filtered into four two-dimensional component images having differing spatial frequency characteristics. The LL component image corresponds to the component image representing the low frequency

spatial information. The LH component image emphasizes horizontal high frequency information. such as horizontal edges. The HL component image emphasizes vertical high frequency information, and the HH component image emphasizes high frequency information related to edges at 45 degrees and 135 degrees.

As noted above, human observers place more importance on the low frequency information, i.e., the LL component image. If only this component image were retained, and the pixels thereof coded using the same number of bits per pixel as used to code the original frame of the motion picture, an image compression of 4:1 would be obtained. Unfortunately, 4:1 compression ratios are insufficient for most purposes. Further, as will be discussed in more detail below, retention of at least some of the high frequency information is needed to maintain picture quality.

To increase the compression ratios, the transformation described with reference to FIG. 2 is iterated on the LL image as illustrated in FIG. 3. It will be assumed that the original image 72 has $P \times P$ pixels. Image 72 is decomposed into four component images, 74-77, as described above, component image 74 being the LL image. Each of the component images 74-77 are $(P/2) \times (P/2)$ pixels.

Next, component image 74 is decomposed into its four component images 80-83, component image 80 being the LL component image. Component images 80-83 each have $(P/4) \times (P/4)$ pixels. The iteration continues by decomposing the LL component into its component images at each stage. Hence, at the next stage, component image 80 will be decomposed into component images 84-87. The iteration is continued until the LL image is reduced to some predetermined size.

The frequency characteristics of the various component images will depend on the specific filter coefficients used. In addition, the frequency characteristics will depend on the level of the decomposition at which the component image was generated. For example, component images 75-77 will in general represent higher frequency information than component images 81-83 which in turn will represent higher frequency information than component images 85-87.

It should be noted that the memory space needed to store the image at each stage of the decomposition is the same as that needed to store the original image. While the device shown in FIG. 2 utilized separate memory arrays for the various component images at the different stages of the transformation, it will be apparent to those skilled in the art that the outputs of analyzer 44 can be placed back in the same array used to store the original frame pixel array provided the outputs in question are appropriately buffered.

Once the single frames making up a given three-dimensional image have been filtered as described above, the filtered two-dimensional component images are combined to generate a set of three-dimensional component images representing the three-dimensional image. An apparatus for generating the three-dimensional filtered component images is shown in FIG. 4 at 100. Coding apparatus 100 receives a sequence of frames 101 which provide the input to a two-dimensional filter 102. Filter 102 operates iteratively on each frame to generate a set of two-dimensional filtered images in a manner analogous to that described above with reference to FIGS. 2 and 3. The output of filter 102 for each frame is stored in a time frame buffer. Exemplary time frame buffers are shown at 104-107. The component images in each time frame buffer are stored such that corresponding pixels are stored at like addresses. A sequence of such pixels is shown at 111-114.

Each sequence of corresponding pixels in the time frame buffers is treated as a one-dimensional digital signal. The digital signal is inputted to a one-dimensional filter 121 which is similar to analyzer 44 shown in FIG. 2. The output of filter 121 comprises the pixels of high and low frequency filtered three-dimensional images. These pixels are stored in a three-dimensional buffer 122. Filter 121 may be applied iteratively to the contents of buffer 122 to provide three-dimensional component images with different degrees of compression in the time-domain in a manner analogous to that described with reference to FIG. 4 above. While a separate buffer 122 is shown for storing the three-dimensional component images, it will be apparent to those skilled in the art that the three-dimensional component images may be stored back in the time component images, may be stored back in the time frame buffers in a manner analogous to that described above with reference to the buffers shown in FIG. 2.

The amount of memory needed to store the three-dimensional component images is determined by the number of taps in filter 121, i.e., the value of N in Eq. (1), and the number of times the time transformation is iterated. If filter 121 is only applied once, i.e., the transformation is not iterated, the number of frames needed for the three-dimensional image will be N. Since the memory need for storing the three-dimensional component images represents a significant fraction of the hardware expense. time-domain transformations with a small N value are preferred. The smallest value of N is 2.

In the preferred embodiment of the present invention, the filtration and quantizing operations are carried out by a pipe-lined processor. The operation of such a processor 200 will now be explained in detail with reference to FIG. 5. The individual frames 201 of a motion picture are received by image compressor 200 one frame at a time. Each time a frame is received from the motion picture, the frame is immediately filtered in two dimensions to form a set of two-dimensional filtered component images. The two-dimensional component images are stored in a time buffer 204. It will be assumed that the time domain filter requires N_f frames from the motion picture to construct the three-dimensional filtered picture. N_f will in general be the number of a(k) values in the one dimensional time domain filter. Hence, the number of time domain buffers must be at least N_f.

The second stage of the pipeline requires groups of N_f frames as input. When the two-dimensional component images from N_f frames have been accumulated, the controller 206 causes the one-dimensional time filter 208 to generate the three dimensional component images 210. In the preferred embodiment of the present invention, these component images are stored in the time buffers in which the two-dimensional component images were stored. However, to clarify the present discussion, a separate buffer is shown in FIG. 5. During this process, incoming frames that have been transformed in two dimensions are stored in one or more additional time buffers. The number of additional time buffers depends on the time needed to complete the time domain filtering. quantize the pixels of the filtered three-dimensional component images, and transmit the quantized pixels on the communication link. There must be sufficient buffers to hold all of the two-dimensional

filtered component images generated during these operations.

The third stage of the pipeline computes the statistics of the pixel groups to be quantized. In the preferred embodiment of the present invention, controller 206 includes data specifying which pixels are to be examined for possible quantization. In general, the pixels may be viewed as comprising three groups. One group will always be quantized. The low-frequency component image is such a group. The low-frequency component image is typically quantized to a predetermined accuracy that is set independent of the statistics of this component image. This accuracy is sufficiently high to guarantee that quantization errors in this component image will be negligible.

The second group is always ignored. That is, zero bits are allocated to the pixels of this group. With the exception of the high frequency component image discussed below with reference to horizontally scanned and interlaced data, the highest frequency component images are typically ignored.

Finally, there are groups of component images that may be quantized if the group has sufficient power in the component image. Here, the power in a component image is defined to be the sum of the squares of the pixel values for that image. This last class may also include portions of a component image. In the larger component images, it is sometimes advantageous to divide the component image into blocks which are quantized separately. These larger component images represent relatively high frequency information. Only those portions of the component image corresponding to edges in the original image are significant. If the entire component image were to be quantized, a large number of bits would be devoted to the regions having no useful information. If, however, only those regions having a significant number of non-zero pixels, or some other discernable feature such as a line, are quantized, this waste can be reduced. The number of bits allocated to this group of component images depends on the statistics of the pixels in each component image and frequencies represented by each component image. As noted above, higher frequency component images may be quantized utilizing less than the statistically optimum number of bits.

Once the three-dimensional component images have been quantized, the information specifying the quantization transformation is sent over the communication link followed by the quantized pixels. In the preferred embodiment of the present invention, the quantization information is sent first to enable the receiver to reverse the quantization of the pixels as the quantized pixels are received.

The above described embodiments of the present invention have been discussed with reference to monochromatic motion pictures. The manner in which color motion pictures are compressed will now be discussed. Color may be specified either in a RGB format or in a luminance/chrominance format. In the RGB format three frames are specified for each frame of the motion picture, one giving the intensity of red, one giving the intensity of blue, and one giving the intensity of green. In principle, each of these frames may be separately compressed and transmitted over the communication link and decompressed at the receiver. However, the amount of compression obtained is significantly less than that obtained utilizing the preferred embodiment of the present invention and a luminance/chrominance format.

In a luminance/chrominance format, each frame is also represented by three frames. The first frame represents the overall light intensity associated with each pixel of the frame. The remaining two frames, referred to as the chrominance frames, provide color signals which when combined with the luminescence frame generate the RGB frames described above. It is found experimentally that significantly more compression can be applied to the chrominance frames than to the luminance frame. Viewed separately, the chrominance frames would appear of a much poorer quality than the luminance frame; however, when recombined to form RGB frames at the receiver, the lower quality of the color chrominance frames is not apparent to a human observer. As a result, significantly greater overall compression can be accomplished while preserving the apparent image quality of the motion picture. Hence, the preferred embodiment of the present invention operates on a luminance frame and two chrominance frames. If the motion picture is not in this format, appropriate conversion circuitry is incorporated to generate the luminance and chrominance frames from the RGB frames. The manner in which the conversion is accomplished is conventional in the photographic arts, and hence, will not be described in more detail here.

The present invention also makes use of the relationships between the data in the two chrominance frames to further compress the chrominance data. In general, the range of colors in an image is considerably less than the maximum range that would be deduced from the variations in the chrominance images. Images are typically made up of objects of relatively uniform colors. Consider a uniformly colored object. There will be corresponding uniform intensity objects in each of the chrominance frames in the same locations. Hence, the data in the two chrominance frames will be highly correlated in the regions occupied by the object in question. This correlation may be used to further reduce the number of bits needed to store the chrominance information.

FIG. 6 illustrates the compression of a motion picture comprising luminance frames 401 and first and second chrominance frames 402 and 403 by a compression apparatus 400 according to the present invention. The luminance frames 401 are compressed in a manner analogous to that described above with reference to FIG. 5. As each luminance frame 401 is received, a two-dimensional filter 410 decomposes the frame into two-dimensional component images. The component images corresponding to each received luminance frame are stored in the next available frame buffer 414. When a predetermined number of luminance frames have been so processed, a time domain filter 418 converts the two-dimensional component images to a plurality of three-dimensional component images which are stored in a buffer 420. The three-dimensional component images in buffer 420 are then quantized by quantizer 422 to form a luminance output signal 424.

The first and second chrominance frames 402 and 403 corresponding to the luminance frames received by filter 410 are processed by two-dimensional filters 411 and 421 in an analogous manner to generate two-dimensional chrominance component images which are stored in frame buffers 424 and 434. In the preferred embodiment of the present invention, only the low-frequency component images which are shown at 425 and 435 are

actually generated and subjected to the subsequent processing. When the above mentioned predetermined number of frames have been received, time-domain filters 419 and 439 convert the two-dimensional chrominance component images to three-dimensional component images which are stored in buffers 428 and 448. In the preferred embodiment of the present invention, only the low-frequency chrominance three-dimensional component images shown at 417 and 437 are actually generated and subjected to the subsequent processing.

The degree of compression applied to the first chrominance frames is the same as that applied to the second chrominance frames; hence, the component images stored in buffers 428 and 448 have corresponding pixels. Each pair of corresponding pixels may be considered to be components of a two-dimensional vector. That is, a single vector valued component image is created from each of the corresponding component images in buffers 428 and 448. These vector valued component images are then quantized by vector quantizer 438 to form a chrominance output signal 444. Because of the above-mentioned correlation between the first and second chrominance frames, the number of vector values actually realized in the motion picture is considerably less than the product of the numbers of values realized in each chrominance frame. As a result, the number of bits needed to represent the chrominance data is significantly reduced.

Before describing the manner in which motion pictures compressed by the present invention are decompressed, the manner in which interlaced motion picture images are preferably compressed will be described. An interlaced motion picture may be regarded as a motion picture with frames having half the number of horizontal scan lines in which every other frame is offset by half the line spacing in the vertical direction. In effect, each frame fills in the spaces between the lines of the previous frame. In principle, each of the frames may be treated as individual frames of the motion picture and compressed as described above. However, a significant improvement in compression can be obtained if the frames are combined to form a single frame having twice the number of scan lines which is then compressed.

The amount of compression that may be obtained is dependent on the amount of redundant information in a frame. There is considerably more redundant information in successive frames of an interlaced sequence than in a single frame; hence, improved compression may be obtained. Unfortunately, it is found experimentally that merely combining two successive frames and compressing the combined frame leads to unacceptable artifacts. In particular, certain fine-grained background which is stationary in the original motion picture sequence appears to vibrate back and forth in the decompressed motion picture. It has been found experimentally that this artifact can be alleviated by preferentially quantizing certain high frequency information that would otherwise not be quantized at all or at most only with limited accuracy.

Consider the two-dimensional decomposition of a frame into two-dimensional filtered component images as described above with reference to FIG. 3. After decomposition, the two-dimensional component images are organized into a hierarchy as shown in FIG. 7. The first level of decomposition generates three high frequency component images shown at 230-232. Component image 230 emphasizes high frequency information relating to horizontal edges, component image 232 emphasizes high frequency information relating to vertical edges, and component image 231 emphasizes high frequency information related to diagonal edges. Normally, these component images would be discarded or at most certain blocks of pixels therein quantized to limited precision.

When the image consists of interlaced scan data, component image 230 corresponding to edges in the horizontal direction, i.e., the direction of the scan in the interlaced data, is selectively filtered and a portion of the data retained. Referring to FIG. 8, component image 230 is filtered in two-dimensions in a manner analogous to the filtering of the first low-frequency component image 234 shown in FIG. 6. That is, component image 230 is input to a two-dimensional filter and the resulting four component images stored back in the same portion of the two-dimensional array as originally used for storing component image 230. The resulting four component images are shown at 240-243 of FIG. 7. The process is iterated on the low-frequency component image generated by each filtration pass. At the completion of the process, the original image will have been decomposed into two hierarchical sets of component images and the original high frequency component images shown at 231 and 232 in FIG. 7.

In the preferred embodiment of the present invention, component images 231 and 232 are not quantized at all. These component images are discarded. The component images derived from component image 230 and the high-frequency component images derived from component image 234 are quantized to a precision that depends on the variance of the particular component image and the original component image from which it was derived. The component images derived from component image 230 are quantized to a precision which is less than the precision that would be dictated by an optimum quantization algorithm. The high-frequency component image derived from component image 234 are quantized at the optimum level. The low-frequency component image derived from component image 234 is quantized to a much higher precision than that dictated by an optimum quantization algorithm.

The motion pictures compressed by the compression method of the present invention can be decompressed by reversing the quantization steps and then combining the various filtered three-dimensional component images utilizing the inverse of the FIR filter banks used to generate the component images. Since the inverse filter banks combine two component images to form a single image, the inverse filter banks will be referred to as synthesizers.

In the preferred embodiment of the present invention, the synthesis is accomplished by iteratively applying one-dimensional synthesizers. A one dimensional synthesizer operates on the high and low frequency vectors generated by the conjugate mirror filter bank described above to form a signal vector whose high and low frequency components are the high and low frequency vectors in question. If the components of the low and high frequency vectors are denoted by $x_L(i)$ and $x_H(i)$, respectively, the synthesizer generates an output vector having components z(i). The number of components in the output vector is twice that in each of the input vectors. The output vector is calculated as follows:

$$x(2m) = \sum_{k=0}^{N/2-1} a(2k)x_L[m-(N/2)-k] - \sum_{k=0}^{N/2-1} a(2k-1)x_H(m-k) \quad (5)$$

$$x(2m-1) = \sum_{k=0}^{N/2-1} a(2k-1)x_L[m-(N/2)-k) - \sum_{k=0}^{N/2-1} a(2k)x_H(m-k) \quad (6)$$

Here, the a(k) are the filter coefficients used by the filter bank that generated the high and low frequency vectors x_H and x_L.

As noted above, the decompression process is begun by first reversing the quantization transformation. If a particular component image was discarded prior to quantization, the component image in question is replaced by a component image of the appropriate size in which all of the pixel values are zero. Since the various component images were quantized to differing degrees of precision, the resultant filtered three-dimensional component images are only an approximation to the three-dimensional component images generated from the sequence of frames received by the transmitter.

As apparatus for decompressing the motion picture is shown at **300** in FIG. **9**. Approximations to three-dimensional component images are generated from the quantized data received by dequantizer **302** which receives data from the bandwidth limited communication channel **303**. The three-dimensional component images are stored in a buffer **304**. The filtering transformation are reversed using the appropriate synthesizers to generate a sequence of frames which approximates the frames used to construct the original three-dimensional image. First, a one-dimensional time domain synthesizer is used to generate a plurality of two-dimensional filter component images which are stored in time buffers **306**. The time buffers can be the same as the buffers used to store the three-dimensional component images. However, for clarity, separate buffers are shown. Each time buffer **306** stores the two-dimensional filtered component images needed to generate one frame of the motion picture after the time domain synthesizer has acted. The contents of each buffer are then iteratively passed through synthesizer **310** which combines the various component images into a single two-dimensional image which is outputted from apparatus **300**.

Apparatus **300** is preferably constructed as a pipelined processor. In the first stage of the processor, dequantization takes place and the three-dimensional filtered component images generated. The second stage of the processor reverses the time domain transformation. The third stage of the processor generates a motion picture frame from each of the sets of two-dimensional component images.

The apparatus shown in FIG. **9** may be replicated for the purpose of decompressing color images that have been compressed by the method described above. In this case, the apparatus shown in FIG. **9** would be used to decompress the luminace image. The version used to decompress the chrominance images would contain two such apparatuses and a dequantizer that unpacks the quantized vector data placing the different components of the vectors in the buffers analogous to buffer **304** described with reference to FIG. **9**.

It should be noted that additional compression of the quantizer output in the present invention may be obtained by using conventional arithmetic coding schemes to eliminate any redundancy in the quantizer output. Such schemes replace repeated sequences of values by codes requiring fewer bits than the sequence in question. It is to be understood that the quantizers and dequantizers described above may also include such coding schemes. Vector coding schemes are also known to those skilled in the art and may be applied to the vectors generated from the chrominance component images.

As pointed out above, the bandwidth limited communication channel may include some means for storing the compressed motion picture such as magnetic tape or computer disk storage. Hence, the present invention may be utilized to generate a recording of a motion picture in a compressed format. Such recordings may be duplicated in the compressed format and disseminated to end users equipped with the appropriate decompression apparatuses.

Such a recording would include means such as a magnetic or optical storage medium for storing a plurality of digital values. The digital values would be divided into groups in which each group would include approximations for the pixels of one or more component images, each such component image specifying a three-dimensional image. Each of the three dimensional image would include information specifying two or more frames of the motion picture. The component images according to this embodiment of the present invention would include a low-frequency component image and one or more high-frequency component images. In the preferred embodiment of this recording, each group would also include digital values specifying two-dimensional chrominance vectors as described above. If the recording had been derived from an interlaced scanning device, each group would also include a high frequency component image emphasizing horizontal image features in the frames of said motion picture.

There has been described herein a novel apparatus and method for compressing and decompressing motion pictures. Various modifications to the present invention will become apparent to those skilled in the art from the foregoing description and accompanying drawings. Accordingly, the present invention is to be limited solely by the scope of the following claims.

What is claimed is:

1. An apparatus for compressing a motion picture comprising a sequence of frames, each said frame comprising a two-dimensional image representing said motion picture at one instance in time, said apparatus comprising:

means for receiving a signal specifying a compression ratio for said motion picture;

means for receiving signals specifying said frames;

means for storing a predetermined number, N_f of consecutively received said frames;

three-dimensional filtering means for filtering said stored frames to generate a plurality of three-dimensional component images, said plurality of three-dimensional component images comprising a low-frequency component image and one or more high-frequency component images, each said component image comprising a plurality of pixel values; and

quantization means for approximating each pixel values in at least one of said component images by an approximation comprising one of a predetermined

set of integer values, the precision of said integers being related to the statistical distribution of the pixel values in said component image and said received compression ratio; and

means for generating output signals specifying said pixel approximations.

2. The motion picture compression apparatus of claim 1 wherein said quantization means comprises means for approximating the pixels of said low-frequency image to a precision greater than the statistically optimum precision determined by said statistical distributions of said pixels in said component images and said received compression ratio; and means for approximating said pixels of at least one of said high frequency component images to a precision less than the statistically optimum precision determined by the statistical distributions of said pixels in said component images and said received compression ratio.

3. The motion picture compression apparatus of claim 1 wherein said frames of said motion picture comprise interlaced frames, every other said frame being offset by one half the horizontal line spacing of said frames, wherein said receiving means comprises means for combining each pair of consecutively received frames to form a single frame having twice the number of horizontal lines, wherein said three-dimensional filtering means comprises means for generating at least one high frequency component image emphasizing horizontal edges in said frames of said motion picture, and wherein said quantization means comprises means for approximating the pixels of said high frequency component image emphasizing horizontal edges to a precision greater than said statistically optimum precision determined by the statistical distribution of said pixels in said high-frequency component image emphasizing horizontal edges.

4. The motion picture compression apparatus of claim 1 wherein each said frame of said motion picture comprises a luminance frame and first and second chrominance frames, wherein said three dimensional filtering means further comprises first and second three-dimensional chrominance filtering means, said first three-dimensional chrominance filtering means comprising means for filtering N_f of said first chrominance frames to generate a first set of three-dimensional filtered chrominance component images, said second three-dimensional chrominance filtering means comprising means for filtering the N_f said second chrominance frames corresponding to N_f of said first chrominance frames filtered by said first three-dimensional chrominance filtering means to generate a second set of three-dimensional filtered chrominance component images, and wherein said quantization means further comprises vector quantization means for approximating corresponding pixels in said first and second three-dimensional filtered chrominance component images by vectors of a predetermined precision.

5. The motion picture compression apparatus of claim 1 wherein said receiving means comprises:

two-dimensional filtering means for filtering each of said received frames to generate a two-dimensional low-frequency component image and one or more two-dimensional high frequency component images, each said component image comprising a plurality of pixel values, wherein

said storing means comprises a plurality of frame buffers, each said frame buffer comprising means for storing said component images generated from one of said consecutively received frames, and wherein

said three-dimensional filtering means comprises a one-dimensional filtering means for combining corresponding pixels in said frame buffers.

6. The motion picture compression apparatus of claim 5 wherein said two-dimensional filtering means comprises a one-dimensional conjugate mirror filter bank.

7. An apparatus for decompressing a compressed motion picture, said compressed motion picture comprising a sequence of approximations for the pixels of one or more three-dimensional images, said decompression apparatus comprising:

means for receiving a signal specifying said pixel approximations;

means for generating one or more three-dimensional component images from said received pixel approximations including means for generating blank pixel values for any said pixel value not received;

inverse three-dimensional filtering means for combining said three-dimensional component images to generate a plurality of two-dimensional luminance picture frames;

means for generating output signals specifying said generated luminance picture frames;

8. The motion picture decompression apparatus of claim 7 wherein said pixel approximations comprise a plurality of two dimensional vector values, the components of each said vector value specifying first and second chrominance pixel values, wherein said means for generating three-dimensional component images comprises means for assembling said first and second chrominance pixel values into first and second sets of chrominance component images, and wherein said inverse three-dimensional filtering means further comprises means for combining said first and second sets of chrominance component images into first and second two-dimensional chrominance picture frames, and wherein said output signal generating means further comprises means for outputting said chrominance picture frames.

9. A method for compressing a motion picture to a given compression ratio, said motion picture comprising a sequence of frames, each said frame comprising a two-dimensional image representing said motion picture at one instance in time, said apparatus comprising:

receiving signals specifying said frames;

storing a predetermined number N_f of consecutively received said frames;

filtering said stored frames to generate a plurality of three-dimensional component images, said plurality of three-dimensional component images comprising a low-frequency component image and one or more high-frequency component images, each said component image comprising a plurality of pixel values; and

approximating each pixel values in at least one of said component images by an approximation comprising one of a predetermined set of integer values, the precision of said integers being related to the statistical distribution of the pixel values in said component image and said compression ratio; and

generating output signals specifying said pixel approximations.

10. The method of claim 9 wherein said approximating step comprises approximating the pixels of said low-frequency image to a precision greater than the statistically optimum precision determined by said statistical distributions of said pixels in said component

images and said compression ratio; and approximating said pixels of at least one of said high frequency component images to a precision less than the statistically optimum precision determined by the statistical distributions of said pixels in said component images and said compression ratio.

11. The method claim 9 wherein said frames of said motion picture comprise interlaced frames, every other said frame being offset by one half the horizontal line spacing of said frames, wherein said receiving step comprises combining each pair of consecutively received frames to form a single frame having twice the number of horizontal lines, wherein said three-dimensional filtering step comprises generating at least one high frequency component image emphasizing horizontal edges, and wherein said approximating step comprises approximating the pixels of said high frequency component image emphasizing horizontal edges to a precision greater than said statistically optimum precision determined by the statistical distribution of said pixels in said high-frequency component image emphasizing horizontal edges.

12. The method of claim 9 wherein each said frame of said motion picture comprises a luminance frame and first and second chrominance frames, wherein said filtering step further comprises filtering N_f of said first chrominance frames to generate a first set of three-dimensional filtered chrominance component images, filtering the N_f said second chrominance frames corresponding to said N_f said first chrominance frames to generate a second set of three-dimensional filtered chrominance component images, and wherein said approximating step approximating corresponding pixels in said first and second three-dimensional filtered chrominance component images by vectors of a predetermined precision.

13. The method claim 9 wherein said receiving step comprises:

filtering each of said received frames to generate a two-dimensional low-frequency component image and one or more two-dimensional high frequency component images, each said component image comprising a plurality of pixel values, wherein

said storing step comprises storing said two-dimensional component images in a plurality of frame buffers, each said frame buffer comprising means for storing said component images generated from one of said consecutively received frames, and wherein

said filtering step comprises combining corresponding pixels in said frame buffers.

14. The method of claim 13 wherein said two-dimensional filtering step comprises filtering said pixels with a one-dimensional conjugate mirror filter bank.

15. A recording of a motion picture comprising a plurality of ordered frames, said recording comprising means for storing a plurality of digital values, said digital values being divided into groups, each said group comprising approximations for the pixels of one or more component images, said component images specifying a three-dimensional image, each said three dimensional image comprising information specifying two or more frames of said motion picture, said component images comprising a low-frequency component image and one or more high-frequency component images.

16. The recording of claim 15 wherein each said group further comprises digital values specifying two-dimensional chrominance vectors, said chrominance vectors specifying first and second three-dimensional chrominance component images, said first and second chrominance component sub-images specifying a first and second three-dimensional chrominance images.

17. The recording of claim 15 wherein each said group comprises a high frequency component image emphasizing horizontal image features in the frames of said motion picture.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,121,191

DATED : June 9, 1992

INVENTOR(S) : Cassereau, *et al.*

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Title page:

Item, [56] *Attorney, Agent, or Firm* - delete "Bartles" and insert therefor --Bartels--.

Item, [73] Assignee - delete "Combridge" and insert therefor --Cambridge--.

Column 2, line 52, after "amount" insert --equal--.

Column 12, lines 19 and 20, delete "maybe stored back in the time component images,".

Column 17, line 28, delete "As" and insert therefor --An--.

[57] Abstract - delete "derrived" and insert therefor --derived--.

Column 2, line 55, after "as" insert --a--.

Column 10, line 57, delete "Analyzer" and insert therefor --analyzer--.

Column 18, line 27, delete "image" and insert therefor --images--.

Column 18, line 67, delete "values" and insert therefor --value--.

Column 20, line 25, delete ";" and insert therefor --.--.

Column 20, line 56, delete "values" and insert therfor --value--.

Column 21, line 7, after "method" insert --of--.

Signed and Sealed this

Twenty-eighth Day of September, 1993

Bruce Lehman

Attest:

BRUCE LEHMAN

Attesting Officer

Commissioner of Patents and Trademarks